

ĐỒ ÁN CƠ SỞ

ATM SIMULATOR SYSTEM

Ngành: **Công nghệ thông tin**

Chuyên ngành: **Công nghệ phần mềm**

Giảng viên hướng dẫn : Trần Đăng Khoa

Sinh viên thực hiện : Phan Hồng Yên Quỳnh

MSSV: 2080600991 Lớp: 20DTHE2

TP. Hồ Chí Minh, 2023

LỜI CAM ĐOAN

Tôi xin cam đoan rằng đồ án cơ sở lập trình java với đề tài mô phỏng ATM (ATM Simulator System) là thực hiện bởi chính tôi. Đồng thời chức năng mà tôi đã làm được, đều nhờ tìm hiểu trên mạng, đọc tài liệu và thực hiện trong quá trình học môn học, hoàn toàn trung thực, không sao chép từ bất kỳ một công trình nghiên cứu khác nào. Những tài liệu trích dẫn đều đã được ghi rõ nguồn gốc.

Tôi xin chịu hoàn toàn trách nhiệm trước tất cả các thầy cô nếu trường hợp phát hiện ra bất cứ sai phạm hay vấn đề sao chép nào trong đề tài này.

Mục Lục

<i>LỜI CAM ĐOAN</i>	1
<i>Mục Lục</i>	2
<i>Danh mục các bảng</i>	5
<i>Danh mục hình vẽ</i>	6
<i>Chương 1: TỔNG QUAN</i>	9
1.1 Giới thiệu đề tài.....	9
1.2 Nhiệm vụ và cấu trúc đồ án.....	13
1.2.1 Mục đích nghiên cứu đồ án	13
1.2.2 Tầm quan trọng đồ án	13
1.2.3 Phạm vi nghiên cứu đồ án.....	13
1.2.4 Nhiệm vụ	14
1.2.5 Cấu trúc đồ án.....	14
<i>Chương 2: CƠ SỞ LÝ THUYẾT</i>	15
2.1 Tổng quan về Java Swing.....	15
2.1.1 Giới thiệu Java Swing	15
2.1.2 Các thành phần của Java Swing	16
2.1.3 Sử dụng Java Swing trong phát triển ứng dụng	17
2.2 Giới thiệu MySQL Server.....	19
2.2.1 Định nghĩa MySQL Server	19
2.2.2 Các tính năng của MySQL Server	20
2.2.3 Cách sử dụng MySQL Server trong ứng dụng.....	20
2.3 Mã hoá RSA.....	21
2.3.1 Định nghĩa và lịch sử của RSA	21
2.3.2 Các bước mã hoá và giải mã RSA	22
2.3.3 Ứng dụng mã hoá RSA trong ATM Simulator System	23
2.4 Mã hoá SHA-512	23

2.4.1 Định nghĩa và lịch sử của SHA-512	23
2.4.2 Các bước mã hoá SHA-512	24
2.4.3 Ứng dụng mã hoá SHA-512 trong ATM Simulator System.....	25
2.5 Hệ thống quản lý SMS cấp cao Twilio	25
2.5.1 Giới thiệu về Twilio	25
2.5.2 Các tính năng của Twilio	26
2.5.3 Sử dụng Twilio trong ATM Simulator System.....	26
2.6 Một số phần mềm hỗ trợ khác.....	27
2.6.1 IntelliJ.....	27
2.6.2 Github.....	28
2.6 Phân tích thiết kế.....	29
2.6.1 Mô tả chức năng.....	29
2.6.2 Bảng phân tích thiết kế chức năng ATM	29
2.6.3 Sơ đồ tổng thể chức năng (Use Case)	31
2.7 Mô hình Activity Diagram	39
2.7.1 Mô hình Login Activity Diagram	39
2.7.2 Mô hình Rút tiền Activity Diagram	40
2.7.3 Mô hình Chuyển khoản Activity Diagram.....	41
2.8 Phân tích thiết kế CSDL.....	42
2.8.1 Xây dựng mô hình ER Diagram.....	42
2.8.2 Mô tả mối liên kết CSDL	42
2.9 Mô hình Class Diagram	45
Chương 3: KẾT QUẢ THỰC NGHIỆM	49
3.1 Giao diện đăng nhập	49
3.2 Giao diện đăng ký	52
3.3 Giao diện tổng quát – atm	55
3.4 Giao diện rút tiền.....	56
3.5 Giao diện gửi tiền.....	63

3.6 Giao diện chuyển tiền	66
3.7 Giao diện kiểm tra số dư tài khoản	71
3.8 Giao diện in sao kê.....	71
3.9 Giao diện đổi mã pin	73
<i>Chương 4: KẾT LUẬN VÀ KIẾN NGHỊ</i>	78
4.1. Kết quả đạt được	78
4.2 Đánh giá sản phẩm.....	78
4.3 Hướng phát triển và mở rộng đề tài	78
<i>TÀI LIỆU THAM KHẢO</i>	80

Danh mục các bảng

BẢNG 2.1: BẢNG PHÂN TÍCH THIẾT KẾ CHỨC NĂNG ATM.....	29
BẢNG 2.2: BẢNG BIỂU DIỄN THUỘC TÍNH SIGNUP.....	46
BẢNG 2.3: BẢNG BIỂU DIỄN THUỘC TÍNH SIGNUP TWO.....	47
BẢNG 2.4: BẢNG BIỂU DIỄN THUỘC TÍNH SIGNUP THREE.....	47
BẢNG 2.5: BẢNG BIỂU DIỄN THUỘC TÍNH LOGIN	48
BẢNG 2.6: BẢNG BIỂU DIỄN THUỘC TÍNH ATM	48
BẢNG 2.7: BẢNG BIỂU DIỄN THUỘC TÍNH BANK_ACCOUNT.....	48

Danh mục hình vẽ

HÌNH 1.1: ATM SIMULATOR SYSTEM	9
HÌNH 1.2: RSA ENCRYPTION	10
HÌNH 1.3: SHA-512 ENCRYPTION	11
HÌNH 1.4: SÁCH APPLIED CRYPTOGRAPHY	12
HÌNH 2.1: JAVA SWING	15
HÌNH 2.2: CÁC THÀNH PHẦN JAVA SWING	16
HÌNH 2.3: JAVA SWING TRÊN INTELLIJ IDE	18
HÌNH 2.4: MYSQL	19
HÌNH 2.5: VÍ DỤ CÁCH KẾT NỐI MYSQL	21
HÌNH 2.6: MÃ HOÁ BẰNG RSA	22
HÌNH 2.7: MÃ HOÁ BẰNG SHA-512	24
HÌNH 2.8: SMS TWILIO	25
HÌNH 2.9: TRÌNH BIÊN DỊCH INTELLIJ	27
HÌNH 2.10: GITHUB	28
HÌNH 2.11: SƠ ĐỒ USECASE TỔNG QUÁT	31
HÌNH 2.12: SƠ ĐỒ USECASE PHÂN RÃ ĐĂNG KÝ	32
HÌNH 2.13: SƠ ĐỒ USECASE PHÂN RÃ ĐĂNG NHẬP	33
HÌNH 2.14: SƠ ĐỒ USECASE PHÂN RÃ RÚT TIỀN	34
HÌNH 2.15: SƠ ĐỒ USECASE PHÂN RÃ CHUYỂN KHOẢN	35
HÌNH 2.16: SƠ ĐỒ USECASE PHÂN RÃ THAY ĐỔI MÃ PIN	36
HÌNH 2.17: SƠ ĐỒ USECASE PHÂN RÃ IN SAO KÊ	37
HÌNH 2.18: SƠ ĐỒ USECASE PHÂN RÃ NẠP TIỀN	38
HÌNH 2.19: MÔ HÌNH LOGIN ACTIVITY DIAGRAM	39
HÌNH 2.20: MÔ HÌNH RÚT TIỀN ACTIVITY DIAGRAM	40
HÌNH 2.21: MÔ HÌNH CHUYỂN KHOẢN ACTIVITY DIAGRAM	41
HÌNH 2.22: MÔ HÌNH ERD DIAGRAM	42
HÌNH 2.23: MÔ TẢ MỐI LIÊN KẾT GIỮA BẢNG LOGIN VÀ ATM	43
HÌNH 2.24: MÔ TẢ MỐI LIÊN KẾT GIỮA BẢNG ATM VÀ BANK_ACCOUNT	43

HÌNH 2.25: MÔ TẢ MỐI LIÊN KẾT GIỮA BẢNG SIGNUP VÀ SIGNUPTWO	43
HÌNH 2.26: MÔ TẢ MỐI LIÊN KẾT GIỮA BẢNG SIGNUPTWO VÀ SIGNUPTHREE ..	43
HÌNH 2.27: MÔ TẢ MỐI LIÊN KẾT GIỮA BẢNG SIGNUPTHREE VÀ LOGIN.....	44
HÌNH 2.28: MÔ HÌNH CLASS DIAGRAM	45
HÌNH 3.1: GIAO DIỆN ĐĂNG NHẬP VÀO ATM.....	49
HÌNH 3.2: THÔNG BÁO NHẬP SAI MÃ PIN	50
HÌNH 3.3: THÔNG BÁO TÀI KHOẢN BỊ KHOÁ	50
HÌNH 3.0.4: THÔNG BÁO SMS.....	51
HÌNH 3.5: THÔNG BÁO SMS.....	51
HÌNH 3.6: MẪU ĐIỀN THÔNG TIN SỐ 1	52
HÌNH 3.7: MẪU ĐIỀN THÔNG TIN SỐ 2	53
HÌNH 3.8: MẪU ĐIỀN THÔNG TIN SỐ 3	54
HÌNH 3.9: GIAO DIỆN ATM.....	55
HÌNH 3.10: GIAO DIỆN RÚT TIỀN NHANH	56
HÌNH 3.11: GIAO DIỆN RÚT TIỀN	57
HÌNH 3.12: THÔNG BÁO SỐ DƯ TÀI KHOẢN KHÔNG ĐỦ	58
HÌNH 3.13: : KIỂM TRA THÔNG TIN NHẬP	59
HÌNH 3.14: KIỂM TRA SỐ DƯ TÀI KHOẢN.....	60
HÌNH 3.15: THÔNG BÁO RÚT TIỀN THÀNH CÔNG	61
HÌNH 3.16: THÔNG BÁO RÚT TIỀN THÀNH CÔNG	62
HÌNH 3.17: GIAO DIỆN GỬI TIỀN VÀO TÀI KHOẢN	63
HÌNH 3.18: KIỂM TRA THÔNG TIN NHẬP	64
HÌNH 3.19: THÔNG BÁO NẠP TIỀN THÀNH CÔNG	65
HÌNH 3.20: GIAO DIỆN CHUYÊN TIỀN	66
HÌNH 3.21: KIỂM TRA THÔNG TIN NHẬP	67
HÌNH 3.22: KIỂM TRA SỐ DƯ TÀI KHOẢN.....	68
HÌNH 3.23: KIỂM TRA TÀI KHOẢN.....	69
HÌNH 3.24: THÔNG BÁO CHUYÊN TIỀN THÀNH CÔNG.....	70
HÌNH 3.25: GIAO DIỆN KIỂM TRA SỐ DƯ	71

HÌNH 3.26: XUẤT PHIẾU IN SAO KÊ.....	73
HÌNH 3.27: GIAO DIỆN ĐỔI MÃ PIN.....	73
HÌNH 3.28: KIỂM TRA THÔNG TIN NHẬP	74
HÌNH 3.29: : KIỂM TRA ĐIỀU KIỆN NHẬP	75
HÌNH 3.30: KIỂM TRA ĐIỀU KIỆN NHẬP	76
HÌNH 3.31: THÔNG BÁO THAY ĐỔI MÃ PIN THÀNH CÔNG	77

Chương 1: TỔNG QUAN

1.1 Giới thiệu về tài

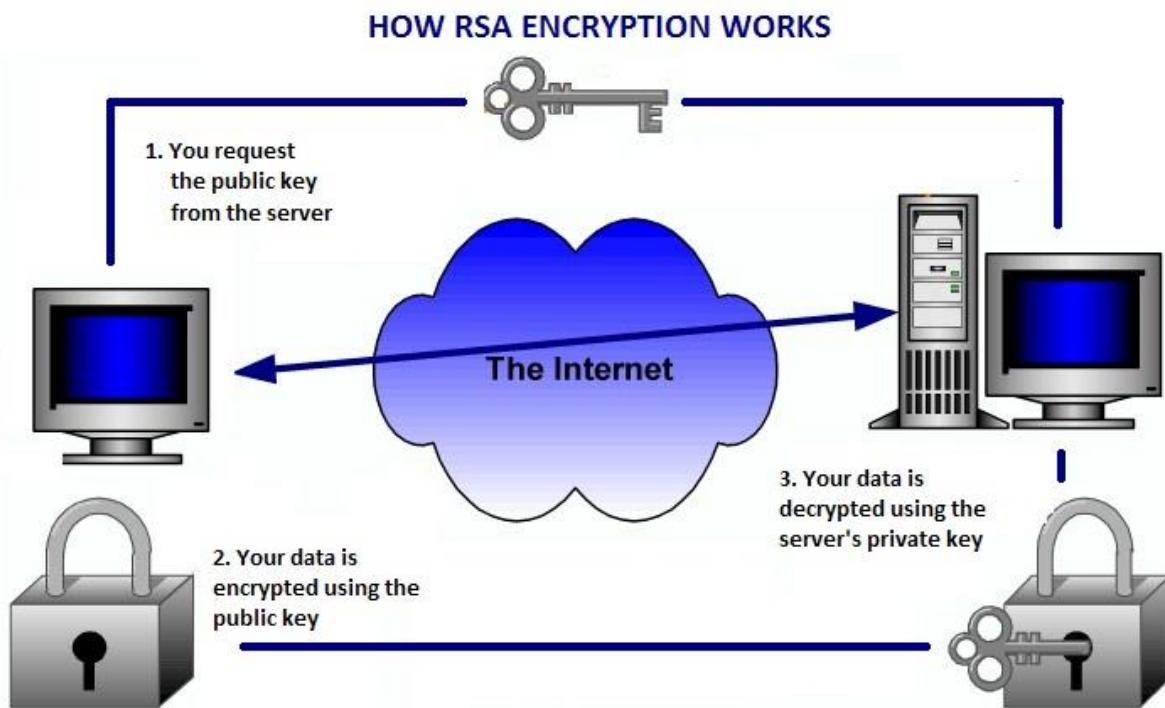
Trong thời đại công nghệ số hiện nay, việc sử dụng các hệ thống thanh toán trực tuyến đang trở thành xu hướng phổ biến và cần thiết. Đặc biệt, việc thực hiện các giao dịch ngân hàng trực tuyến qua các máy ATM là rất quan trọng và tiện lợi cho người dùng. Vì vậy, tôi đã quyết định phát triển một ứng dụng phần mềm mô phỏng các tính năng của các máy ATM thông thường, với mục đích nghiên cứu và giúp người dùng thực hiện các giao dịch ngân hàng một cách nhanh chóng và tiện lợi hơn. Ứng dụng ATM được phát triển bằng ngôn ngữ lập trình Java và sử dụng các công nghệ phổ biến như Java Swing, MySQL Server, hệ thống quản lý SMS Twilio. Đặc biệt đáng chú ý là công nghệ mã hóa RSA và SHA-512.



Hình 1.1: ATM Simulator System

Mã hóa RSA và SHA-512 là hai công nghệ mã hóa được sử dụng trong hệ thống mô phỏng ATM để đảm bảo an toàn cho dữ liệu của người dùng trong quá trình thực hiện các giao dịch. Mã hóa RSA là một thuật toán mã hóa khóa công khai được phát triển bởi Ron Rivest, Adi Shamir và Leonard Adleman vào năm 1977. Thuật toán này

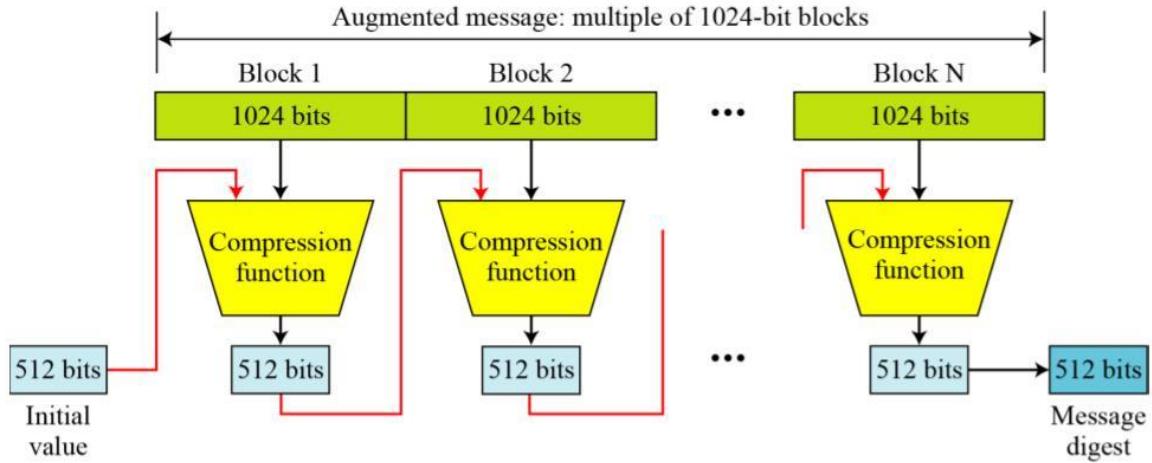
sử dụng hai khóa riêng biệt để mã hóa và giải mã dữ liệu. Khóa công khai được chia sẻ cho mọi người, trong khi khóa bí mật chỉ được giữ bởi chủ sở hữu. RSA được sử dụng trong ATM Simulator System để mã hóa thông tin nhạy cảm của người dùng, chẳng hạn như mật khẩu và mã PIN, đảm bảo rằng chúng không bị lộ khi được truyền qua mạng.



Hình 1.2: RSA Encryption

SHA-512 là một hàm băm mã hóa được sử dụng để tạo ra một mã băm duy nhất cho dữ liệu đầu vào. Điều này đảm bảo tính toàn vẹn của dữ liệu và ngăn chặn việc xâm nhập vào hệ thống thông qua việc thay đổi dữ liệu. SHA-512 được sử dụng trong ATM Simulator System để mã hóa thông tin tài khoản của người dùng và tạo ra một mã băm duy nhất để xác thực dữ liệu.

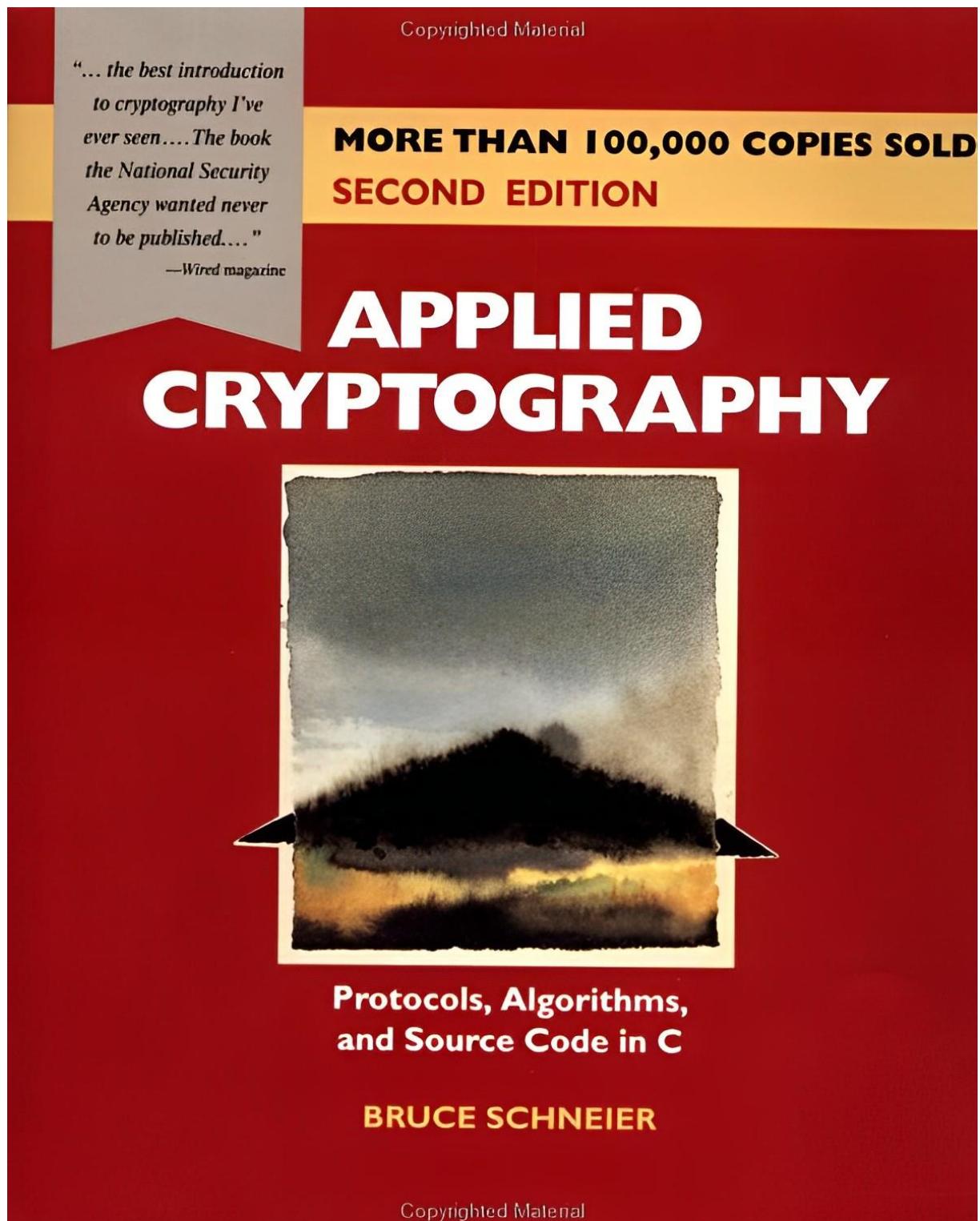
SHA-512 Overview



Hình 1.3: SHA-512 Encryption

Trong quá trình phát triển ATM Simulator System, tôi đã tham khảo nhiều tài liệu và nghiên cứu trước đây liên quan đến các công nghệ mã hóa này. Một số tài liệu tham khảo đáng chú ý bao gồm "Applied Cryptography" của Bruce Schneier, "RSA Encryption Algorithm" của Ron Rivest, Adi Shamir và Leonard Adleman, và "Secure Hash Standard" của National Institute of Standards and Technology (NIST). Các tài liệu này là những tài liệu tham khảo quan trọng trong lĩnh vực mã hóa và được sử dụng để nghiên cứu và phát triển các thuật toán và hệ thống mã hóa. Tài liệu "Applied Cryptography" của Bruce Schneier được coi là một trong những tài liệu cơ bản nhất về lý thuyết và ứng dụng của mã hóa. Tài liệu "RSA Encryption Algorithm" của Ron Rivest, Adi Shamir và Leonard Adleman là tài liệu cơ bản về thuật toán mã hóa RSA, một thuật toán mã hóa khóa công khai phổ biến được sử dụng trong nhiều hệ thống mã hóa. Tài liệu "Secure Hash Standard" của National Institute of Standards and Technology (NIST) là một tiêu chuẩn về hàm băm an toàn, đảm bảo tính toàn vẹn của dữ liệu và được sử dụng trong nhiều hệ thống mã hóa. Tôi đã tham khảo và sử dụng các

tài liệu này trong quá trình phát triển ATM Simulator System để hiểu rõ hơn về cách thức hoạt động của các công nghệ mã hóa RSA và SHA-512 và áp dụng chúng vào hệ thống của tôi.



Hình 1.4: Sách Applied Cryptography

ATM Simulator System là một ứng dụng ngân hàng ảo có tính năng tương tự như các máy ATM thông thường. Người dùng có thể thực hiện các giao dịch như đăng nhập, đăng ký, rút tiền, gửi tiền, chuyển tiền, kiểm tra số dư tài khoản, in sao kê và đổi mã pin một cách dễ dàng và nhanh chóng. Với giao diện đơn giản và thân thiện, ATM Simulator System đảm bảo sự tiện lợi và an toàn cho người dùng.

1.2 Nhiệm vụ và cấu trúc đồ án

1.2.1 Mục đích nghiên cứu đồ án

Mục đích của đồ án là xây dựng một ứng dụng phần mềm mô phỏng các tính năng của các máy ATM thông thường, với mục đích nghiên cứu và giúp người dùng thực hiện các giao dịch ngân hàng một cách nhanh chóng và tiện lợi hơn. Đồ án cũng nhằm mục đích tăng cường kiến thức về các loại giao dịch ngân hàng trực tuyến.

1.2.2 Tầm quan trọng đồ án

Việc phát triển ứng dụng Automated Teller Machine có tầm quan trọng rất lớn đối với cộng đồng, đặc biệt là đối với những người chưa quen với việc sử dụng các hệ thống thanh toán trực tuyến. Nó giúp người dùng tiết kiệm thời gian và công sức trong việc thực hiện các giao dịch ngân hàng và cũng là một công cụ hữu ích để tăng cường kiến thức về các loại giao dịch ngân hàng trực tuyến. Ngoài ra, việc phát triển ứng dụng ATM cũng đóng góp vào sự phát triển của công nghệ và ngành ngân hàng, mở ra những cơ hội mới cho sự phát triển kinh tế.

1.2.3 Phạm vi nghiên cứu đồ án

Đồ án tập trung vào mô phỏng một hệ thống Automated Teller Machine với các tính năng cơ bản đã được liệt kê ở phần trước. Đối tượng nghiên cứu của đồ án là người dùng muốn sử dụng các dịch vụ ngân hàng trực tuyến thông qua máy ATM.

1.2.4 Nhiệm vụ

Nhiệm vụ được đề ra như sau:

- Phân tích và thu thập thông tin về các tính năng cơ bản của hệ thống ATM.
- Sử dụng các công nghệ phổ biến như Java Swing, MySQL Server, mã hóa RSA và SHA-512, hệ thống quản lý SMS Twilio để phát triển hệ thống.
- Thiết kế giao diện người dùng đơn giản và dễ sử dụng.
- Đảm bảo tính an toàn, tính năng và hiệu suất cao của hệ thống.

1.2.5 Cấu trúc đồ án.

Đồ án được chia thành các chương như sau:

- Chương 1: Tổng quan: Giới thiệu đề tài, phạm vi nghiên cứu đồ án, mục đích nghiên cứu đồ án, tầm quan trọng đồ án, nhiệm vụ và cấu trúc đồ án.
- Chương 2: Cơ sở lý thuyết: Tổng quan về Java Swing, giới thiệu về MySQL Server, mã hóa RSA và SHA-512, hệ thống quản lý SMS Twilio, các phần mềm hỗ trợ khác, phân tích thiết kế, mô hình ER Diagram và mô hình cơ sở dữ liệu hoàn thiện.
- Chương 3: Kết quả thực nghiệm: Trình bày về giao diện đăng nhập, giao diện đăng ký, giao diện tổng quát – atm, giao diện rút tiền, giao diện gửi tiền, giao diện chuyển tiền, giao diện kiểm tra số dư tài khoản, giao diện in sao kê và giao diện đổi mã pin.
- Chương 4: Kết luận và kiến nghị: Đánh giá kết quả đạt được, đánh giá sản phẩm, đưa ra hướng phát triển và mở rộng đề tài.

Chương 2: CƠ SỞ LÝ THUYẾT

2.1 Tổng quan về Java Swing

Java Swing là một bộ công cụ giao diện người dùng (GUI) cho ngôn ngữ lập trình Java, cho phép các nhà phát triển xây dựng các ứng dụng desktop đa nền tảng. Swing là một phần của Java Foundation Classes (JFC) và được cung cấp bởi Oracle Corporation như một phần của nền tảng Java SE (Standard Edition).

2.1.1 Giới thiệu Java Swing

Java Swing được phát triển vào những năm 1990 bởi Sun Microsystems, với mục đích tạo ra một bộ công cụ GUI đa nền tảng cho Java. Sau đó, Sun Microsystems được Oracle Corporation mua lại và tiếp tục phát triển Swing.



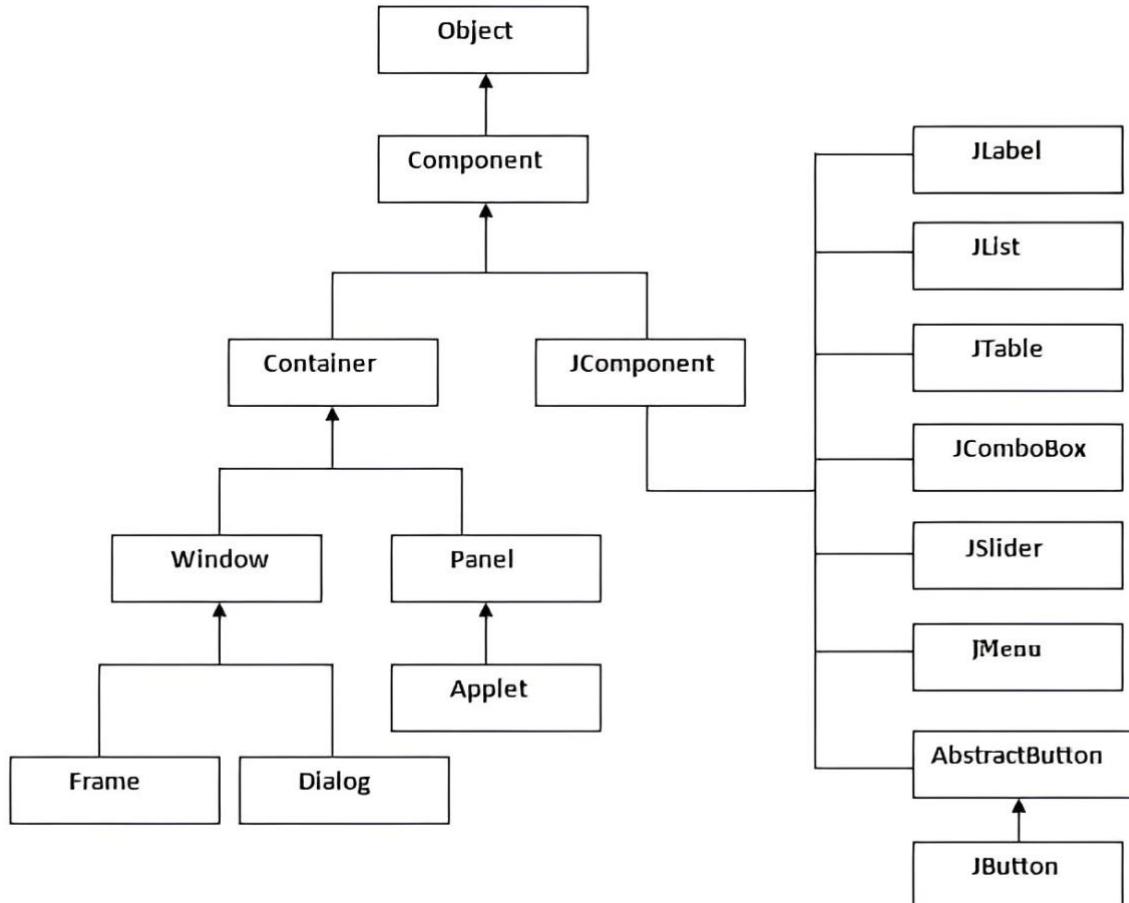
Hình 2.1: Java Swing

Swing cung cấp một loạt các thành phần giao diện người dùng, từ các thành phần cơ bản như nút và hộp văn bản đến các thành phần phức tạp hơn như bảng và cây. Ngoài ra, Swing cũng hỗ trợ các thành phần tùy chỉnh và các tính năng tương tác phong phú.

Swing được viết bằng Java, cho phép phát triển ứng dụng đa nền tảng trên các hệ điều hành khác nhau. Nó cũng được cung cấp với một bộ công cụ phát triển tích hợp (IDE) như NetBeans và IntelliJ IDEA.

2.1.2 Các thành phần của Java Swing

Java Swing cung cấp cho các nhà phát triển một số thành phần giao diện người dùng để tạo ra các ứng dụng đa nền tảng với giao diện đồ họa. Các thành phần này bao gồm:



Hình 2.2: Các thành phần Java Swing

- Containers (các bộ chứa): chứa và quản lý các thành phần khác như buttons, labels, text fields, ...
- Components (các thành phần): cung cấp các đối tượng để xây dựng giao diện người dùng, bao gồm buttons, labels, text fields, checkboxes, radio buttons, menus,...
- Layout managers (các bộ quản lý bố trí): quản lý cách các thành phần được sắp xếp trong một container.

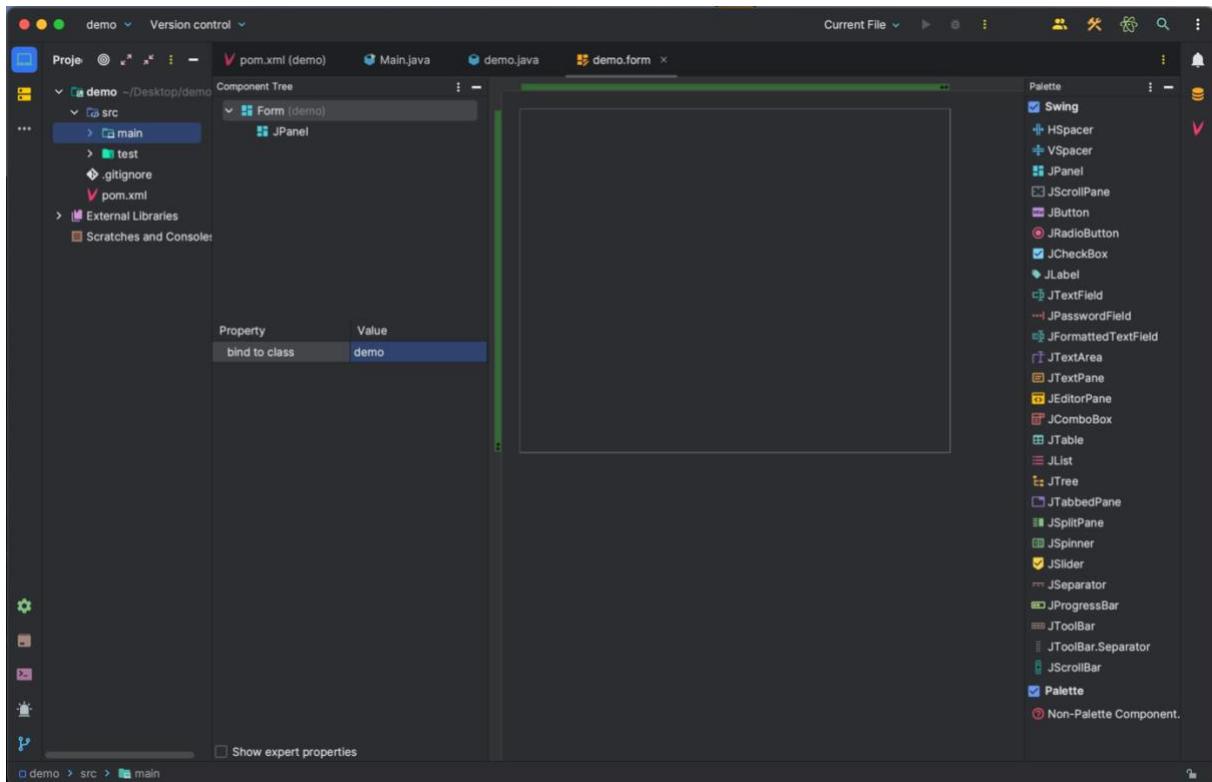
- Events (các sự kiện): cung cấp các cơ chế để xử lý các sự kiện do người dùng tương tác với các thành phần của giao diện.

2.1.3 Sử dụng Java Swing trong phát triển ứng dụng

Để sử dụng Java Swing trong phát triển ứng dụng, các nhà phát triển cần phải tạo các đối tượng Swing components, bố trí chúng trong các container và quản lý các sự kiện được kích hoạt bởi người dùng. Các bước phát triển ứng dụng Java Swing bao gồm:

- Xác định các thành phần cần thiết: Các nhà phát triển cần xác định các thành phần giao diện người dùng cần thiết để phù hợp với yêu cầu của ứng dụng.
- Tạo các đối tượng: Các đối tượng Swing components được tạo ra bằng cách sử dụng các lớp và phương thức trong thư viện Java Swing.
- Bố trí các thành phần: Các thành phần được đặt trong các container để tạo thành giao diện người dùng. Các container được chọn dựa trên cách các thành phần được sắp xếp và kích thước của chúng.
- Sử dụng các bộ quản lý bố trí: Các bộ quản lý bố trí được sử dụng để quyết định cách các thành phần được sắp xếp trong các container.
- Xử lý các sự kiện: Các sự kiện được kích hoạt bởi người dùng, chẳng hạn như nhấn nút hoặc nhập dữ liệu vào một ô text field, được xử lý bằng cách sử dụng các cơ chế của Java Swing.

Tổng quan về Java Swing và các thành phần của nó cung cấp cho các nhà phát triển một bộ công cụ mạnh mẽ và đầy đủ để phát triển các ứng dụng giao diện người dùng cho các ứng dụng Java. Việc sử dụng Java Swing giúp cho các nhà phát triển có thể tạo ra các ứng dụng giao diện đa nền tảng, có tính di động cao, có khả năng tương tác với người dùng và có tính linh hoạt trong thiết kế.



Hình 2.3: Java Swing trên IntelliJ IDEA

Việc sử dụng Java Swing trong phát triển ứng dụng có thể được thực hiện bằng cách sử dụng các IDE như NetBeans, Eclipse hoặc IntelliJ IDEA. Các IDE này cung cấp cho các nhà phát triển một môi trường phát triển tích hợp đầy đủ và hỗ trợ cho việc phát triển ứng dụng Java Swing, bao gồm các công cụ thiết kế giao diện, xây dựng các đối tượng và bố trí chúng trong các container.

Trong quá trình phát triển ứng dụng, các nhà phát triển nên tập trung vào việc thiết kế giao diện người dùng một cách đơn giản và dễ sử dụng, tạo ra các thành phần giao diện được tối ưu hóa cho trải nghiệm người dùng và tăng cường tính năng và hiệu suất của ứng dụng.

2.2 Giới thiệu MySQL Server

2.2.1 Định nghĩa MySQL Server



Hình 2.4: MySQL

MySQL Server là một hệ quản trị cơ sở dữ liệu mã nguồn mở, được sử dụng rộng rãi trên toàn thế giới. MySQL Server có khả năng quản lý và lưu trữ các dữ liệu theo cấu trúc và cung cấp các tính năng như tạo bảng, thêm, sửa, xóa, truy vấn dữ liệu, đảm bảo tính toàn vẹn và an toàn của dữ liệu.

2.2.2 Các tính năng của MySQL Server

MySQL Server cung cấp một số tính năng quan trọng sau:

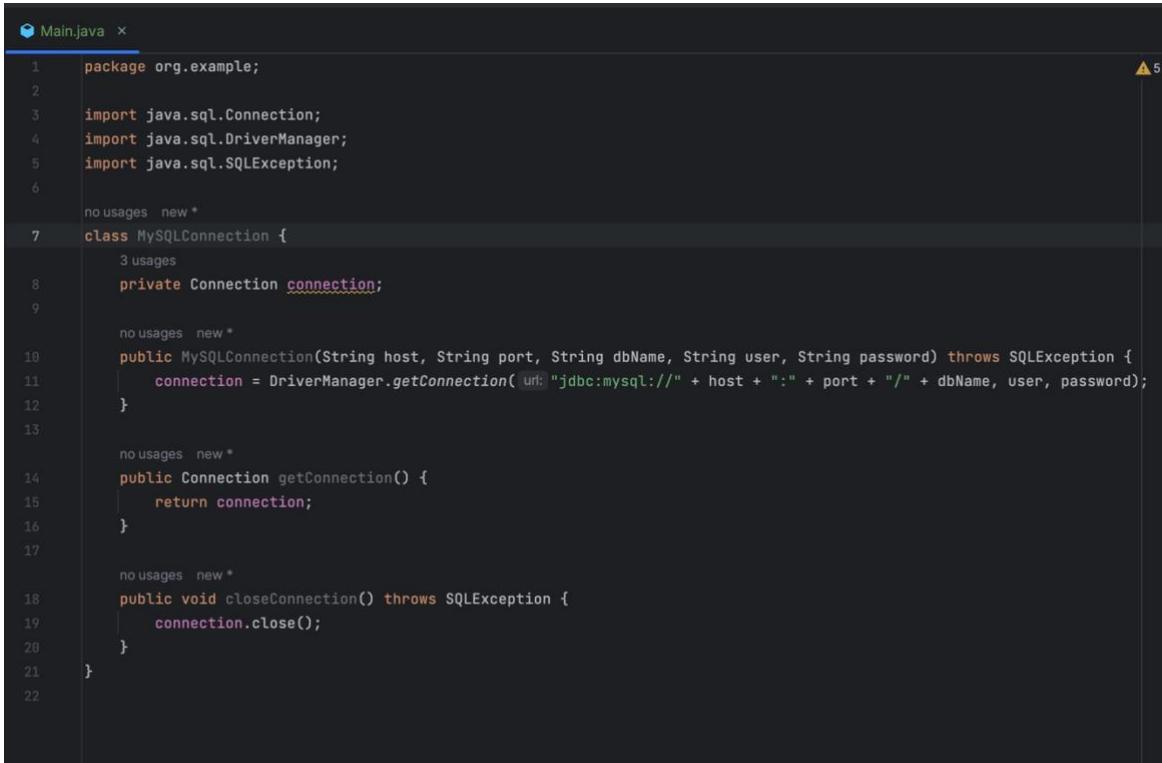
- **Khả năng quản lý dữ liệu:** MySQL Server cho phép lưu trữ và quản lý dữ liệu theo cấu trúc với các đối tượng như bảng, trườn, khóa chính, khóa ngoại và chỉ mục.
- **Tính năng truy vấn dữ liệu:** MySQL Server cung cấp các lệnh truy vấn dữ liệu để truy cập và xử lý dữ liệu trong các bảng và trườn, bao gồm SELECT, INSERT, UPDATE và DELETE.
- **Tính năng tạo và xóa bảng:** MySQL Server cho phép tạo và xóa các bảng dữ liệu để lưu trữ các dữ liệu liên quan và quản lý các đối tượng như trườn, khóa chính, khóa ngoại và chỉ mục.
- **Tính năng tạo cơ sở dữ liệu:** MySQL Server cho phép tạo các cơ sở dữ liệu để quản lý các bảng dữ liệu.
- **Tính năng backup và phục hồi:** MySQL Server cung cấp tính năng sao lưu và khôi phục dữ liệu, giúp bảo vệ dữ liệu khỏi các sự cố hệ thống hoặc nguy cơ mất dữ liệu.

2.2.3 Cách sử dụng MySQL Server trong ứng dụng

Để sử dụng MySQL Server trong ứng dụng, các nhà phát triển có thể sử dụng các thư viện JDBC (Java Database Connectivity) để kết nối và truy vấn dữ liệu. Các bước cơ bản để sử dụng MySQL Server trong ứng dụng bao gồm:

1. Tải xuống và cài đặt MySQL Server.
2. Tạo cơ sở dữ liệu và bảng dữ liệu theo yêu cầu của ứng dụng.
3. Kết nối đến cơ sở dữ liệu MySQL Server bằng cách sử dụng các thông tin như tên máy chủ, tên cơ sở dữ liệu, tên người dùng và mật khẩu.
4. Sử dụng câu lệnh SQL để truy vấn dữ liệu từ cơ sở dữ liệu, hoặc thực hiện các thao tác thêm, sửa, xóa dữ liệu trong bảng.
5. Sử dụng JDBC API để thực hiện kết nối cơ sở dữ liệu và thực hiện các câu lệnh SQL.

Ví dụ, để kết nối đến cơ sở dữ liệu MySQL Server, các lập trình viên có thể làm như sau:

A screenshot of a Java code editor showing a file named Main.java. The code defines a class MySQLConnection with methods to get a database connection and close it. It uses JDBC to establish a connection to a MySQL database.

```
1 package org.example;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.SQLException;
6
7 no usages new *
8 class MySQLConnection {
9     3 usages
10    private Connection connection;
11
12    no usages new *
13    public MySQLConnection(String host, String port, String dbName, String user, String password) throws SQLException {
14        connection = DriverManager.getConnection("jdbc:mysql://" + host + ":" + port + "/" + dbName, user, password);
15    }
16
17    no usages new *
18    public Connection getConnection() {
19        return connection;
20    }
21
22    no usages new *
23    public void closeConnection() throws SQLException {
24        connection.close();
25    }
26}
```

Hình 2.5: Ví dụ cách kết nối MySQL

Trong đó, các thông tin như tên máy chủ, cổng, tên cơ sở dữ liệu, tên người dùng và mật khẩu được truyền vào để tạo kết nối. Sau khi kết nối được thiết lập, các câu lệnh SQL có thể được thực thi bằng cách sử dụng JDBC API.

Sử dụng MySQL Server trong ứng dụng giúp các nhà phát triển lưu trữ và quản lý dữ liệu một cách hiệu quả, đồng thời đảm bảo tính bảo mật và an toàn của dữ liệu.

2.3 Mã hoá RSA

2.3.1 Định nghĩa và lịch sử của RSA

RSA là một thuật toán mã hoá khóa công khai được phát triển bởi Ron Rivest, Adi Shamir và Leonard Adleman vào năm 1977. Tên của thuật toán được đặt theo tên đầu của các tác giả. RSA là một trong những thuật toán mã hoá phổ biến nhất hiện nay và được sử dụng rộng rãi trong việc bảo vệ thông tin truyền tải trên mạng.



Hình 2.6: Mã hoá bằng RSA

Thuật toán RSA dựa trên việc sử dụng một cặp khóa: khóa công khai và khóa bí mật. Khóa công khai có thể được chia sẻ công khai và được sử dụng để mã hoá thông tin, trong khi khóa bí mật được giữ bí mật và được sử dụng để giải mã thông tin đã được mã hoá.

2.3.2 Các bước mã hoá và giải mã RSA

Các bước mã hoá và giải mã RSA như sau:

Bước 1: Chọn hai số nguyên tố lớn p và q và tính $n = p * q$

Bước 2: Tính $m = (p-1)*(q-1)$

Bước 3: Chọn một số nguyên tố e sao cho $1 < e < m$ và e không có ước chung với m . e được chọn là khóa công khai.

Bước 4: Tìm số nguyên tố d sao cho $d * e \% m = 1$. d được chọn là khóa bí mật.

Bước 5: Cặp khóa (e, n) được chia sẻ công khai, còn cặp khóa (d, n) được giữ bí mật.

Bước 6: Mã hoá một thông điệp x bằng cách tính $c = x^e \% n$

Bước 7: Giải mã một thông điệp được mã hoá y bằng cách tính $x = y^d \% n$

2.3.3 Ứng dụng mã hóa RSA trong ATM Simulator System

Trong ATM Simulator System, mã hóa RSA được sử dụng để đảm bảo tính an toàn và bảo mật của thông tin giao dịch giữa người dùng và máy chủ. Khi người dùng đăng nhập hoặc thực hiện các giao dịch như rút tiền, gửi tiền, chuyển tiền, đổi mã PIN, thông tin sẽ được mã hóa bằng RSA trước khi được truyền qua mạng.

Các bước thực hiện mã hóa RSA trong mô phỏng hệ thống ATM bao gồm:

1. Tạo cặp khóa RSA: Hệ thống sẽ tạo ra một cặp khóa RSA gồm khóa công khai (public key) và khóa bí mật (private key). Khóa công khai sẽ được phân phối cho các người dùng để mã hóa thông tin giao dịch, trong khi khóa bí mật sẽ được sử dụng để giải mã thông tin này bởi máy chủ.
2. Mã hóa thông tin giao dịch: Khi người dùng thực hiện các giao dịch như rút tiền, gửi tiền, chuyển tiền, đổi mã PIN, thông tin sẽ được mã hóa bằng khóa công khai của người dùng trước khi được truyền qua mạng.
3. Giải mã thông tin giao dịch: Khi thông tin giao dịch được truyền đến máy chủ, nó sẽ được giải mã bằng khóa bí mật của máy chủ để xác thực và xử lý các giao dịch.

Việc sử dụng mã hóa RSA trong mô phỏng hệ thống ATM đảm bảo tính bảo mật và an toàn của thông tin giao dịch giữa người dùng và máy chủ.

2.4 Mã hóa SHA-512

2.4.1 Định nghĩa và lịch sử của SHA-512

SHA-512 (Secure Hash Algorithm-512) là một thuật toán mã hóa băm dữ liệu đối xứng được phát triển bởi Cục An ninh Quốc gia Hoa Kỳ (NSA). SHA-512 được sử dụng để bảo vệ tính toàn vẹn của dữ liệu và được coi là một trong những thuật toán mã hóa băm mạnh nhất hiện nay. SHA-512 là phiên bản cải tiến của SHA-1 và SHA-2, với độ dài băm là 512 bit.

SHA-512 Initial Digest

- Initial values of message digest H_0
- Designed for appearance of “randomness”
 - Created from first 8 primes (2, 3, 5, 7, 11, 13, 17, 19)
 - Take square root
 - Take first 64 digits of fractional part

Message digest

8 words, each of 64 bits = 512 bits

A	6A09E667F3BCC908
B	BB67AE8584CAA73B
C	3C6EF372EF94F828
D	A54FE53A5F1D36F1
E	510E527FADE682D1
F	9B05688C2B3E6C1F
G	1F83D9ABFB41BD6B
H	5BE0CD19137E2179

14

Hình 2.7: Mã hoá bằng SHA-512

2.4.2 Các bước mã hoá SHA-512

Các bước mã hoá SHA-512 bao gồm:

- Bước 1: Chuẩn bị
Đối với thông điệp đầu vào, chúng ta thêm các bit 0 để đảm bảo độ dài của thông điệp là bội số của 1024 bit (độ dài khối mặc định của SHA-512).
- Bước 2: Khởi tạo giá trị băm ban đầu
Giá trị băm ban đầu được khởi tạo bằng các hằng số được định nghĩa sẵn.
- Bước 3: Tính toán giá trị băm
Thuật toán SHA-512 sử dụng một loạt các phép toán thao tác trên các khối thông điệp để tính toán giá trị băm. Các phép toán này bao gồm xoay vòng (rotate), xoay vòng có kết hợp (rotate and XOR), AND, OR, NOT và ADD.
- Bước 4: Xuất giá trị băm
Sau khi tính toán, giá trị băm được xuất ra dưới dạng một chuỗi các bit nhị phân.

2.4.3 Ứng dụng mã hóa SHA-512 trong ATM Simulator System

Trong hệ thống ATM Simulator System, mã hóa SHA-512 được sử dụng để bảo vệ mật khẩu người dùng. Khi người dùng tạo mật khẩu mới hoặc thay đổi mật khẩu hiện tại, mật khẩu sẽ được mã hóa bằng thuật toán SHA-512 trước khi lưu trữ trong cơ sở dữ liệu.

Khi người dùng đăng nhập vào hệ thống, mật khẩu đã nhập sẽ được mã hóa bằng SHA-512 và so sánh với mật khẩu đã được lưu trữ trong cơ sở dữ liệu. Nếu hai giá trị khớp nhau, người dùng được cho phép truy cập vào hệ thống. Nếu không, hệ thống sẽ từ chối truy cập.

Sử dụng mã hóa SHA-512 giúp tăng đáng kể độ bảo mật của hệ thống ATM Simulator System, bởi vì SHA-512 là một trong những thuật toán mã hóa bảo mật cao được sử dụng rộng rãi hiện nay. Nó có khả năng mã hóa các thông tin lên đến 2^{512} bit, cung cấp một mức độ bảo mật rất cao.

2.5 Hệ thống quản lý SMS cấp cao Twilio

2.5.1 Giới thiệu về Twilio



Hình 2.8: SMS Twilio

Twilio là một hệ thống quản lý tin nhắn SMS và cuộc gọi điện thoại cung cấp cho các doanh nghiệp khả năng tương tác với khách hàng thông qua các phương tiện

này. Twilio cho phép các doanh nghiệp tạo, gửi và quản lý các tin nhắn SMS hoặc cuộc gọi điện thoại trực tiếp từ ứng dụng của họ, cung cấp cho khách hàng một kênh giao tiếp trực tiếp và hiệu quả.

2.5.2 Các tính năng của Twilio

Twilio cung cấp nhiều tính năng hữu ích cho các doanh nghiệp, bao gồm:

- Gửi và nhận tin nhắn SMS: Twilio cho phép các doanh nghiệp gửi và nhận tin nhắn SMS trực tiếp từ ứng dụng của họ, cung cấp cho khách hàng một kênh giao tiếp trực tiếp và tiện lợi.
- Tạo và quản lý các cuộc gọi điện thoại: Twilio cho phép các doanh nghiệp tạo và quản lý các cuộc gọi điện thoại trực tiếp từ ứng dụng của họ, cung cấp cho khách hàng một kênh giao tiếp trực tiếp và hiệu quả.
- Tích hợp với các nền tảng khác: Twilio có thể tích hợp với nhiều nền tảng khác nhau, bao gồm Java, .NET, Ruby và Python.
- Quản lý và theo dõi các cuộc gọi điện thoại và tin nhắn SMS: Twilio cung cấp cho các doanh nghiệp các công cụ để quản lý và theo dõi các cuộc gọi điện thoại và tin nhắn SMS, bao gồm các báo cáo chi tiết về hoạt động của hệ thống.

2.5.3 Sử dụng Twilio trong ATM Simulator System

Trong hệ thống ATM Simulator System, Twilio được sử dụng như một hệ thống quản lý SMS để gửi thông báo và mã xác thực OTP đến điện thoại của người dùng. Để sử dụng Twilio trong ứng dụng, các bước cơ bản bao gồm:

1. Đăng ký tài khoản Twilio và cung cấp các thông tin xác thực, bao gồm ACCOUNT_SID và AUTH_TOKEN.
2. Tạo một số điện thoại Twilio để gửi tin nhắn đến các điện thoại khác.
3. Sử dụng Twilio API để tạo tin nhắn SMS và gửi chúng đến điện thoại của người dùng.
4. Khi một người dùng thực hiện một đăng nhập sai quá nhiều lần, hệ thống sẽ sinh ra một gửi tin nhắn thông báo “tài khoản quý khách bị khoá” đến điện thoại của người dùng thông qua Twilio. Việc sử dụng Twilio trong hệ thống ATM

Simulator System giúp tăng tính bảo mật và đảm bảo rằng các giao dịch chỉ được thực hiện bởi người dùng chính.

2.6 Một số phần mềm hỗ trợ khác

2.6.1 IntelliJ

IntelliJ là một phần mềm hỗ trợ phát triển ứng dụng Java tích hợp với các tính năng phong phú như code editor, debugging, refactoring, quản lý dự án và nhiều tính năng khác. Nó được xem là một trong những phần mềm hỗ trợ phát triển Java hàng đầu hiện nay. IntelliJ hỗ trợ cho nhiều nền tảng, bao gồm Windows, MacOS và Linux.

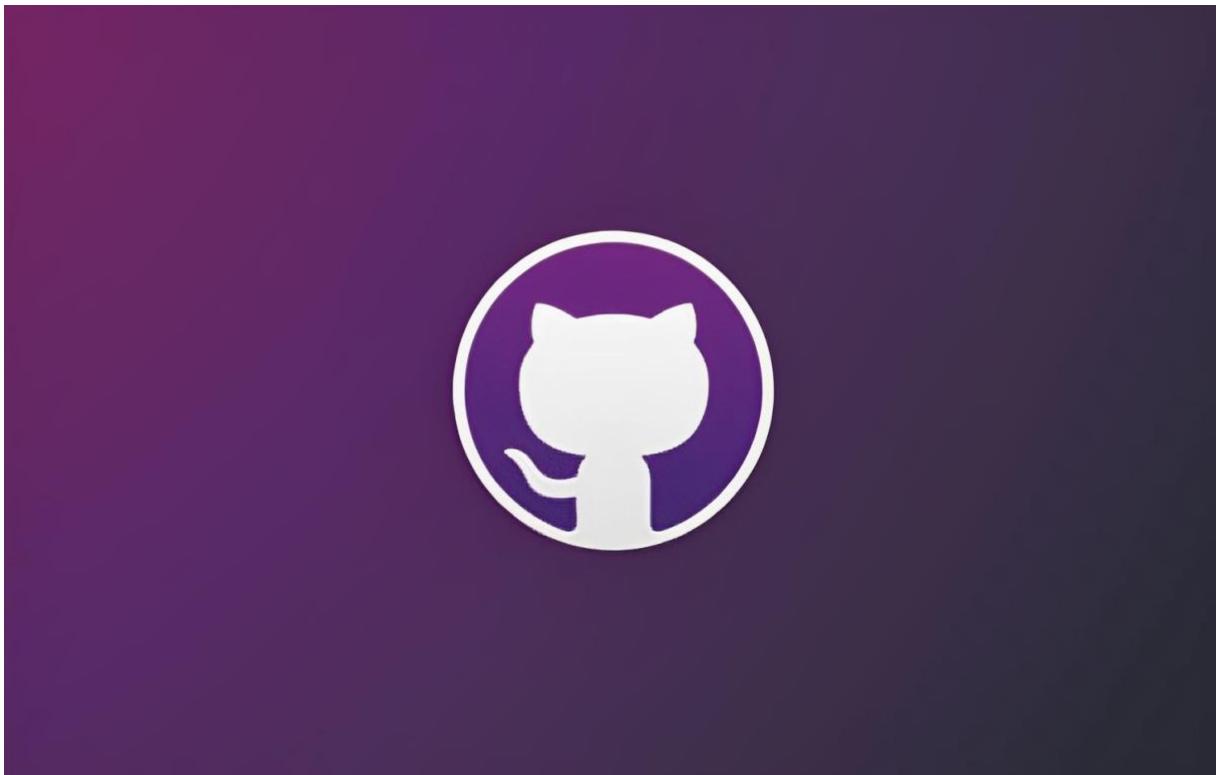


Hình 2.9: Trình biên dịch IntelliJ

Trong đồ án ATM Simulator System, chúng tôi đã sử dụng IntelliJ để phát triển và quản lý dự án. Với IntelliJ, chúng tôi đã dễ dàng tạo, chỉnh sửa và kiểm tra mã nguồn của ứng dụng ATM Simulator System.

2.6.2 Github

Github là một dịch vụ lưu trữ mã nguồn trực tuyến cho phép các nhà phát triển lưu trữ, quản lý và chia sẻ mã nguồn của họ. Github cung cấp các tính năng như quản lý phiên bản, theo dõi thay đổi, hợp tác và nhiều tính năng khác.



Hình 2.10: Github

Trong đồ án ATM Simulator System, chúng tôi đã sử dụng Github để lưu trữ và quản lý mã nguồn của ứng dụng. Nhờ đó, chúng tôi đã có thể làm việc cùng nhau và quản lý các thay đổi trong quá trình phát triển một cách hiệu quả.

2.6 Phân tích thiết kế

2.6.1 Mô tả chức năng

Hệ thống ATM là một trong những cách tiện lợi để thực hiện các giao dịch tài chính, bao gồm rút tiền, xem số dư tài khoản, chuyển khoản, đổi mật khẩu, in thông tin tài khoản và nạp tiền vào tài khoản. Với hệ thống này, người dùng có thể đăng ký tài khoản mới và đăng nhập bằng số thẻ và mã PIN của mình để thực hiện các giao dịch tài chính trực tuyến.

Tuy nhiên, bảo mật và bảo vệ thông tin tài khoản của người dùng là một trong những vấn đề quan trọng nhất mà hệ thống ATM cần đảm bảo. Vì vậy, hệ thống ATM phải được thiết kế và triển khai với các biện pháp bảo mật cao nhằm giảm thiểu nguy cơ mất thông tin tài khoản của người dùng. Các biện pháp này có thể bao gồm kiểm tra số thẻ và mã PIN, đăng nhập an toàn, giới hạn số lần nhập sai mã PIN và khóa tài khoản khi quá số lần cho phép.

2.6.2 Bảng phân tích thiết kế chức năng ATM

Để thiết kế một hệ thống ATM hoạt động hiệu quả và đáp ứng đầy đủ các nhu cầu của người dùng, việc phân tích và thiết kế chức năng cho hệ thống là rất quan trọng. Dưới đây là bảng phân tích thiết kế chức năng cho hệ thống ATM để cung cấp các dịch vụ tài chính cho người dùng:

Bảng 2.1: Bảng phân tích thiết kế chức năng ATM

STT	Nội dung	Mô tả chi tiết	Ghi chú
1	Chức năng đăng ký tài khoản mới	Đăng ký tài khoản mới: Người dùng cung cấp thông tin cá nhân và thông tin tài khoản, hệ thống kiểm tra và tạo tài khoản mới.	
2	Chức năng đăng nhập	Người dùng cung cấp số thẻ và mã PIN, hệ thống kiểm tra và cho phép người dùng truy cập vào tài khoản của mình.	

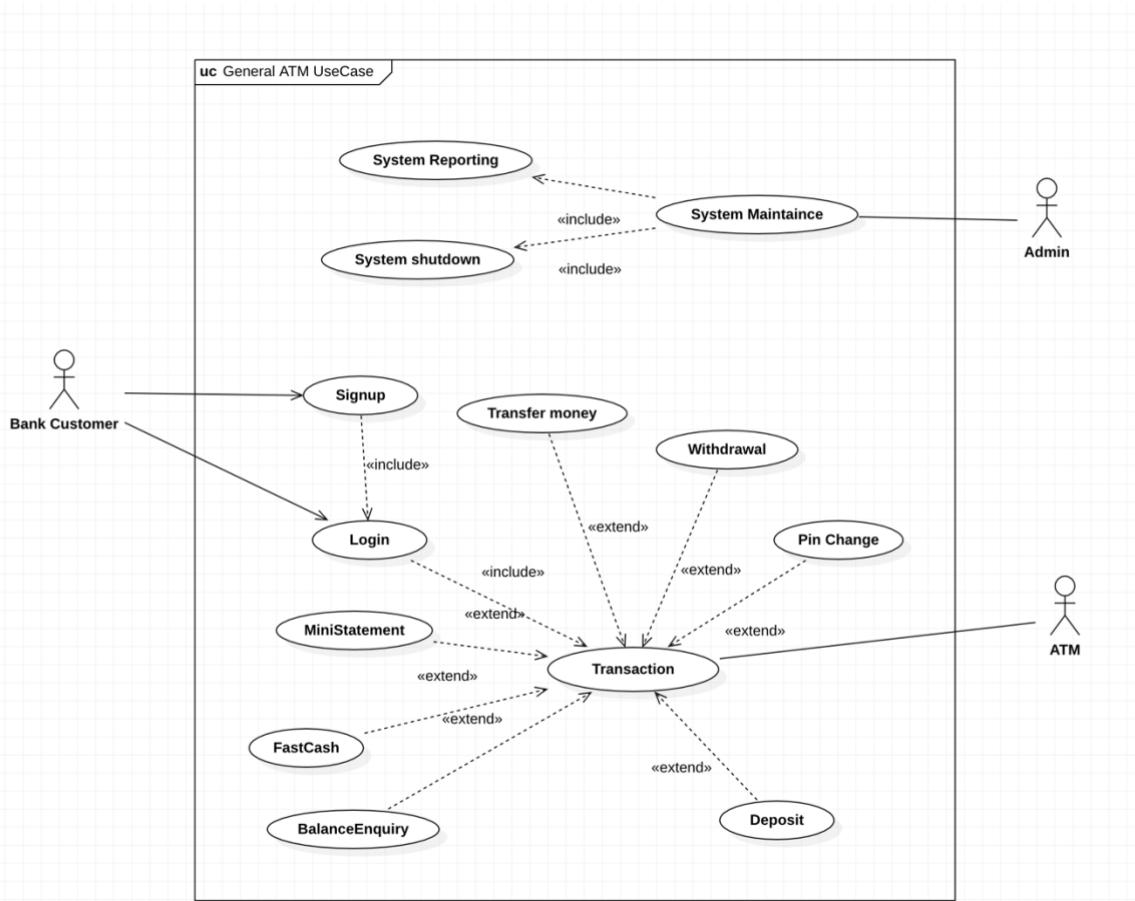
3	Chức năng rút tiền	Người dùng cung cấp số tiền cần rút, hệ thống kiểm tra và thực hiện giao dịch rút tiền.	
4	Chức năng xem số dư tài khoản	Người dùng chọn tùy chọn xem số dư, hệ thống hiển thị số dư tài khoản của người dùng.	
5	Chức năng chuyển khoản	Người dùng cung cấp thông tin tài khoản người nhận và số tiền cần chuyển, hệ thống kiểm tra và thực hiện giao dịch chuyển khoản.	
6	Chức năng đổi mã pin	Người dùng cung cấp mã PIN hiện tại và mã PIN mới, hệ thống kiểm tra và thay đổi mã PIN của người dùng.	
7	Chức năng in sao kê	Người dùng chọn tùy chọn in thông tin lịch sử tài khoản, hệ thống hiển thị thông tin lịch sử tài khoản của người dùng.	
8	Chức năng nạp tiền vào tài khoản	Người dùng cung cấp số tiền cần nạp, hệ thống kiểm tra và thực hiện giao dịch nạp tiền vào tài khoản.	
9	Bảo mật và bảo vệ thông tin	Hệ thống ATM phải có các biện pháp bảo mật và bảo vệ thông tin tài khoản của người dùng, bao gồm kiểm tra số thẻ và mã PIN, đăng nhập an toàn, giới hạn số lần nhập sai mã PIN và khóa tài khoản khi quá số lần cho phép.	

Hệ thống ATM cũng phải hỗ trợ các tính năng khác như:

- Hiển thị thông báo lỗi nếu có lỗi xảy ra trong quá trình thực hiện giao dịch.
- Cập nhật số dư tài khoản ngay sau khi thực hiện giao dịch.
- Thực hiện giao dịch nhanh chóng và đáp ứng các yêu cầu của người dùng một cách chính xác và đáng tin cậy.

2.6.3 Sơ đồ tổng thể chức năng (Use Case)

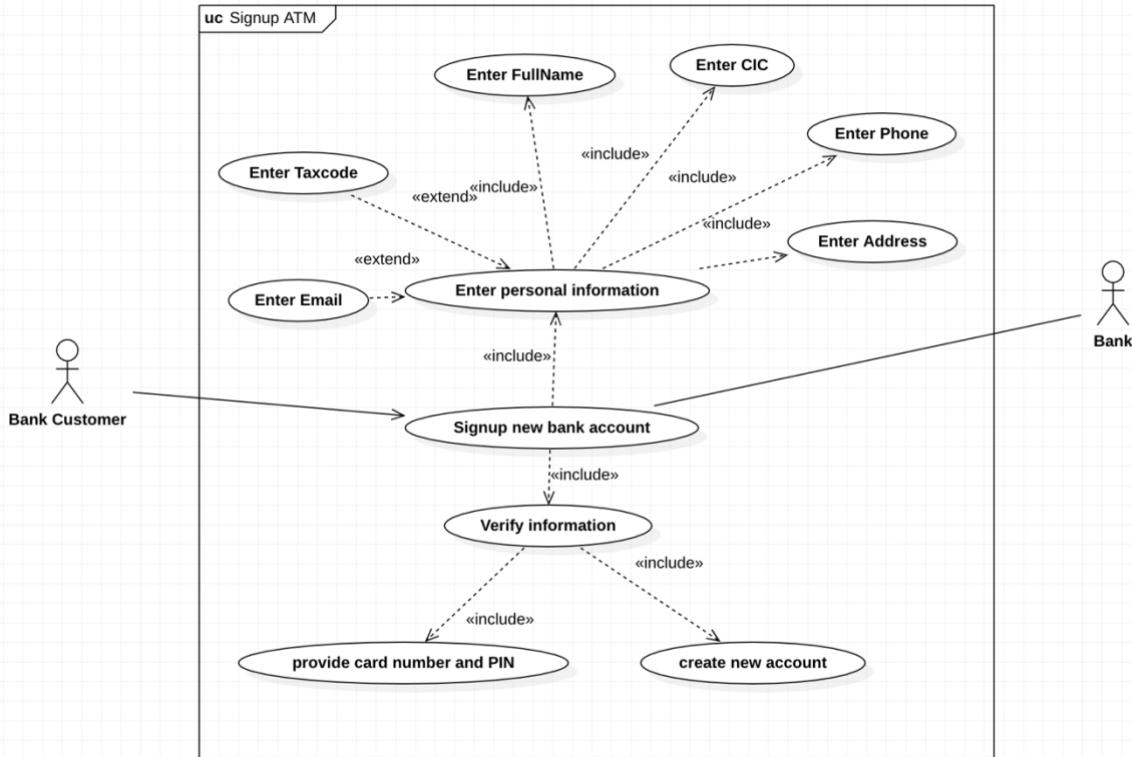
2.6.3.1 Sơ đồ tổng quát Use Case



Hình 2.11: Sơ đồ UseCase Tổng quát

- Người dùng đăng nhập vào hệ thống ATM bằng cách nhập số thẻ và mã PIN. Sau khi đăng nhập thành công, người dùng có thể chọn chức năng cần sử dụng, bao gồm rút tiền, chuyển khoản, đổi mã PIN, in sao kê và nạp tiền. Sau khi hoàn tất chức năng, người dùng có thể đăng xuất khỏi hệ thống ATM.
- Điều kiện trước đó người dùng đã mở hoặc đăng ký thẻ ngân hàng rồi thì mới đăng nhập được vào cây ATM.

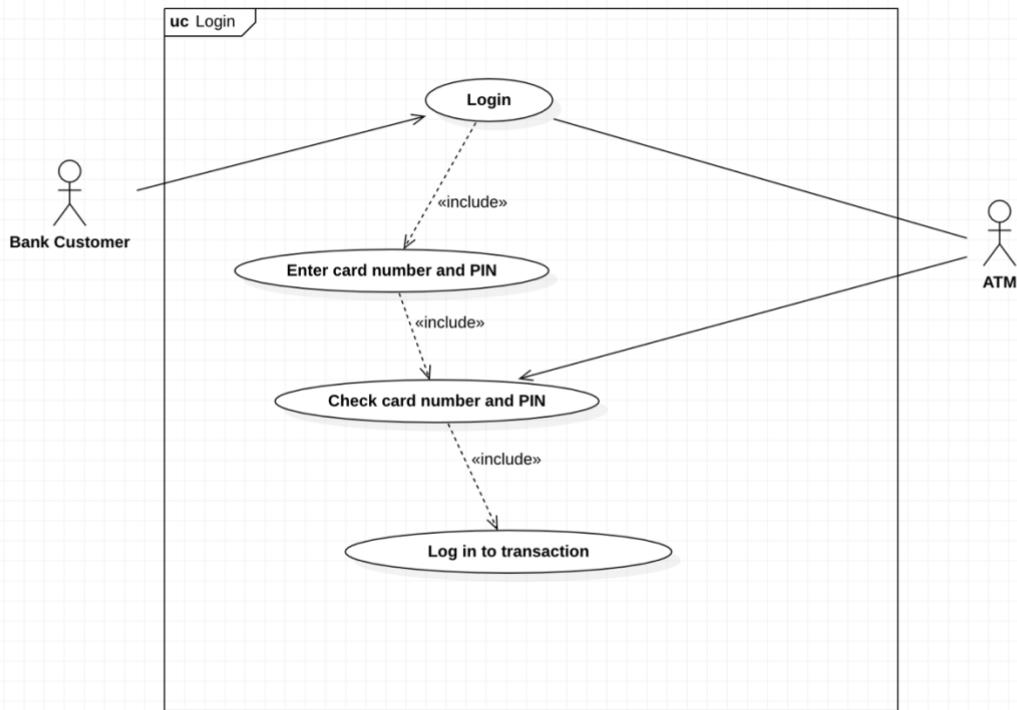
2.6.3.2 Sơ đồ phân rã đăng ký



Hình 2.12: Sơ đồ UseCase phân rã đăng ký

- Khách hàng đến ngân hàng đăng ký tài khoản thẻ tín dụng để có thể truy cập vào hệ thống ATM. Để đăng ký tài khoản khách hàng cần phải điền đầy đủ thông tin vào tờ mẫu đăng ký mà ngân hàng cung cấp cho như bao gồm: họ tên, số điện thoại, căn cước công dân, năm sinh, giới tính,... Thông tin của khách hàng sẽ được ngân hàng kiểm tra và sau đó xác nhận. Sau khi xác nhận xong, ngân hàng sẽ lưu lại thông tin và cấp cho khách hàng thẻ ngân hàng bao gồm có số tài khoản thẻ và mã pin

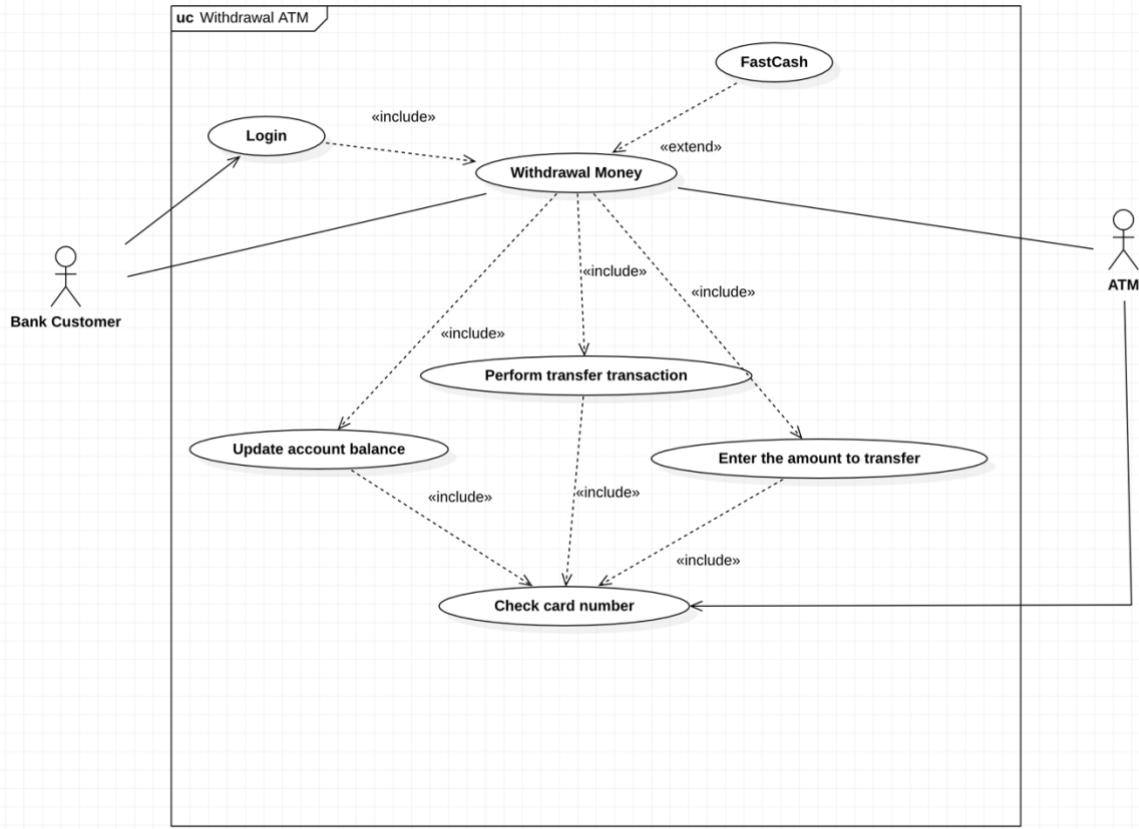
2.6.3.3 Sơ đồ phân rã đăng nhập



Hình 2.13: Sơ đồ UseCase phân rã đăng nhập

- Quá trình đăng nhập bắt đầu khi người dùng nhập số thẻ và mã PIN và kết thúc khi hệ thống truy cập thành công vào giao diện giao dịch. Hệ thống ATM sẽ kiểm tra tính hợp lệ của số thẻ và mã PIN được nhập. Nếu số thẻ hoặc mã PIN không hợp lệ, quá trình đăng nhập sẽ thất bại và hệ thống sẽ yêu cầu người dùng nhập lại thông tin đăng nhập.

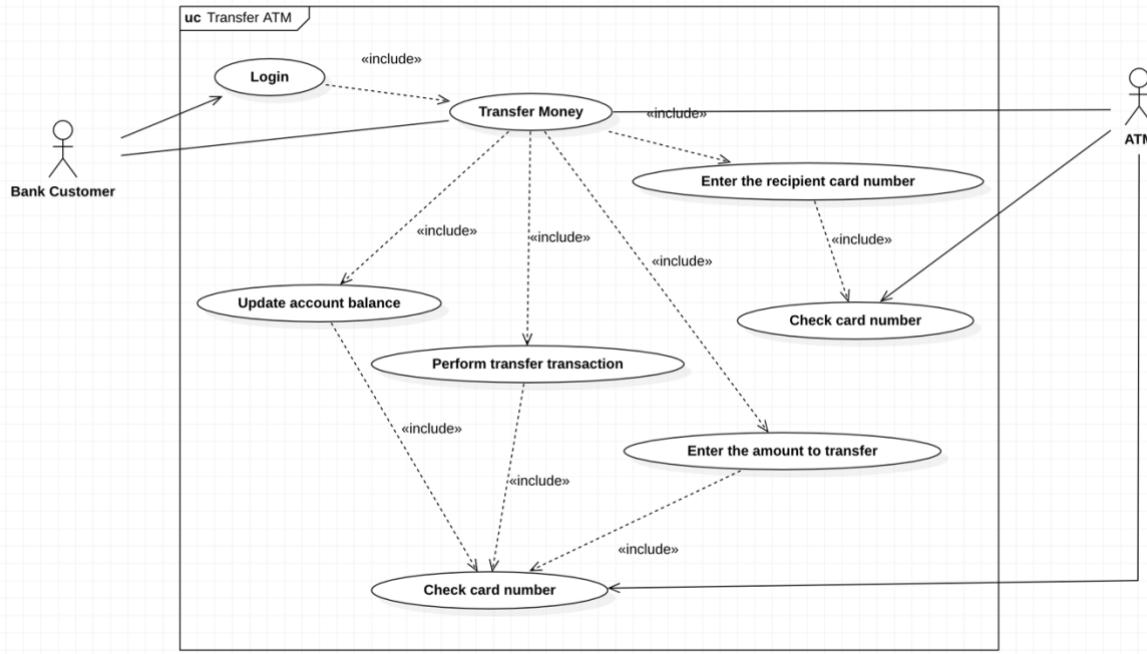
2.6.3.4 Sơ đồ phân rã rút tiền



Hình 2.14: Sơ đồ UseCase phân rã rút tiền

- Sau khi đăng nhập thành công trên ATM. Người dùng lựa chọn chức năng rút tiền trên màn hình ATM, sau đó nhập số tiền cần rút. Hệ thống kiểm tra số dư trong tài khoản người dùng để xác nhận có đủ tiền để rút hay không. Nếu số dư đủ và mã PIN chính xác, hệ thống tiến hành rút tiền từ tài khoản của người dùng và phát ra số tiền tương ứng. Sau khi rút tiền thành công, hệ thống cập nhật số dư tài khoản của người dùng để phản ánh số tiền đã rút. Sau khi hoàn thành rút tiền, hệ thống hiển thị thông báo về kết quả giao dịch và yêu cầu người dùng tiến hành các hành động tiếp theo hoặc kết thúc giao dịch.

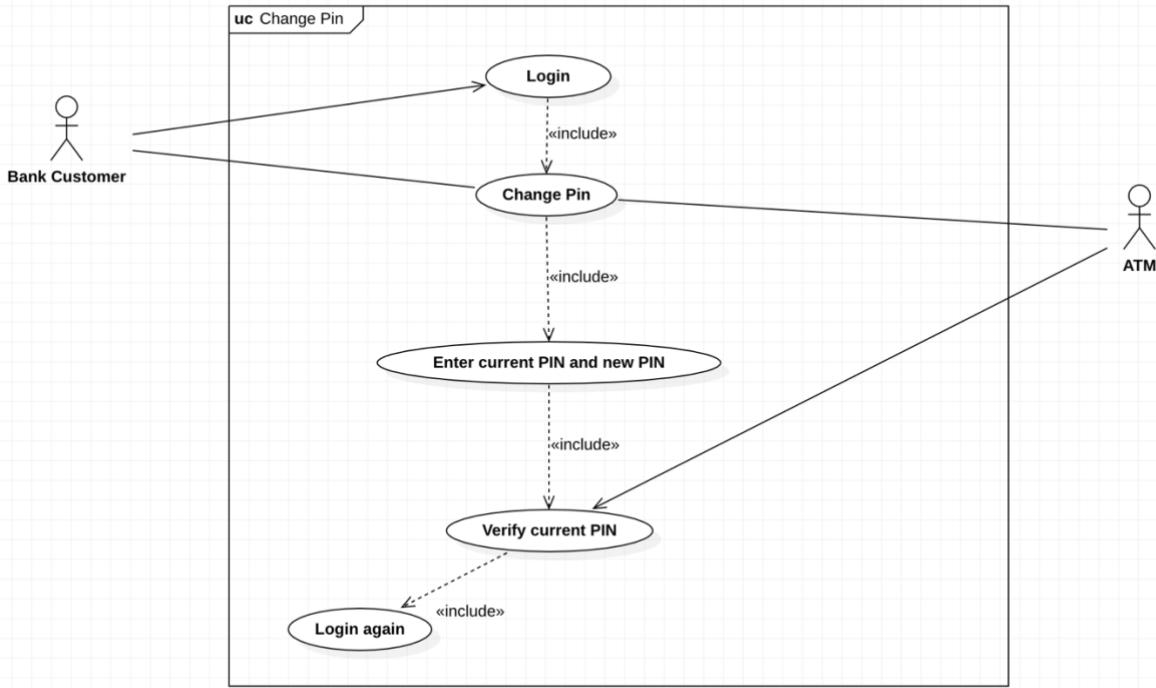
2.6.3.5 Sơ đồ phân rã chuyển khoản.



Hình 2.15: Sơ đồ UseCase phân rã chuyển khoản

- Sau khi đăng nhập thành công trên ATM. Người dùng chọn chức năng Chuyển khoản trên giao diện ATM. Người dùng nhập số tài khoản đích và số tiền cần chuyển. Hệ thống ATM kiểm tra số dư tài khoản nguồn để đảm bảo rằng người dùng có đủ tiền để thực hiện giao dịch. Hệ thống ATM thực hiện giao dịch chuyển khoản bằng cách trừ tiền từ tài khoản nguồn và cộng tiền vào tài khoản đích. Hệ thống ATM cập nhật số dư tài khoản nguồn và tài khoản đích sau khi thực hiện giao dịch.

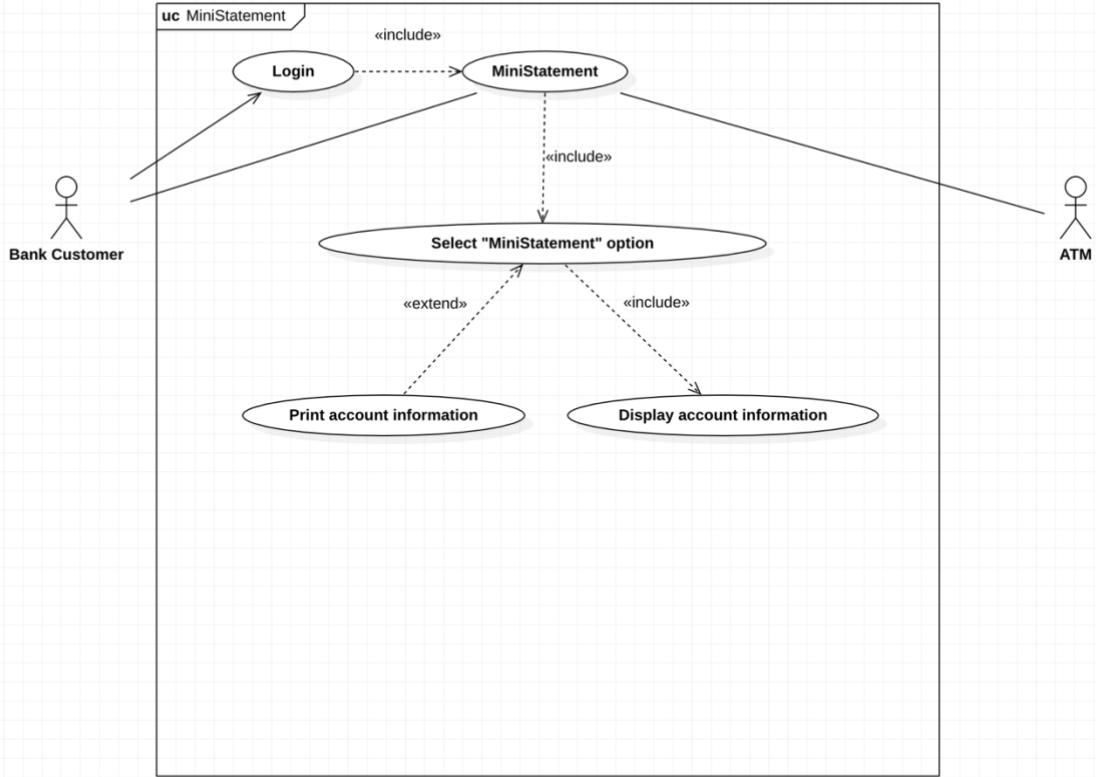
2.6.3.6 Sơ đồ phân rã đổi mật khẩu.



Hình 2.16: Sơ đồ UseCase phân rã thay đổi mã pin

- Sau khi đăng nhập thành công trên ATM. Người dùng chọn chức năng Thay đổi mã pin trên giao diện ATM. Người dùng nhập mã PIN hiện tại và mã PIN mới. Hệ thống kiểm tra xác thực bằng mã PIN hiện tại. Người dùng thay đổi mã PIN. Hệ thống cập nhật lại mã PIN mới của người dùng. Hệ thống ATM thông báo cho người dùng biết thay đổi mã pin đã thành công và yêu cầu đăng nhập lại.

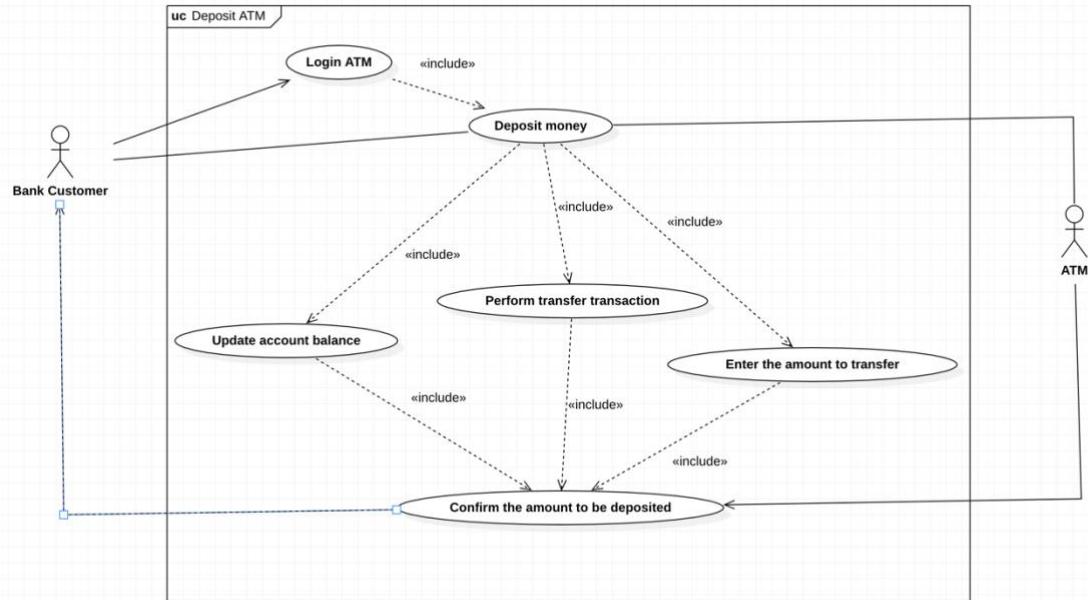
2.6.3.7 Sơ đồ phân rã in sao kê.



Hình 2.17: Sơ đồ UseCase phân rã in sao kê

- Sau khi đăng nhập thành công trên ATM. Người dùng chọn chức năng In sao kê trên giao diện ATM. Hệ thống hiển thị thông tin sao kê tài khoản trên màn hình ATM cho khách hàng xem trước khi quyết định in ra. Nếu khách hàng chọn in sao kê, hệ thống sẽ thực hiện in ra thông tin sao kê tài khoản của khách hàng và đưa cho khách hàng lấy lại.

2.6.3.8 Sơ đồ phân rã nạp tiền.

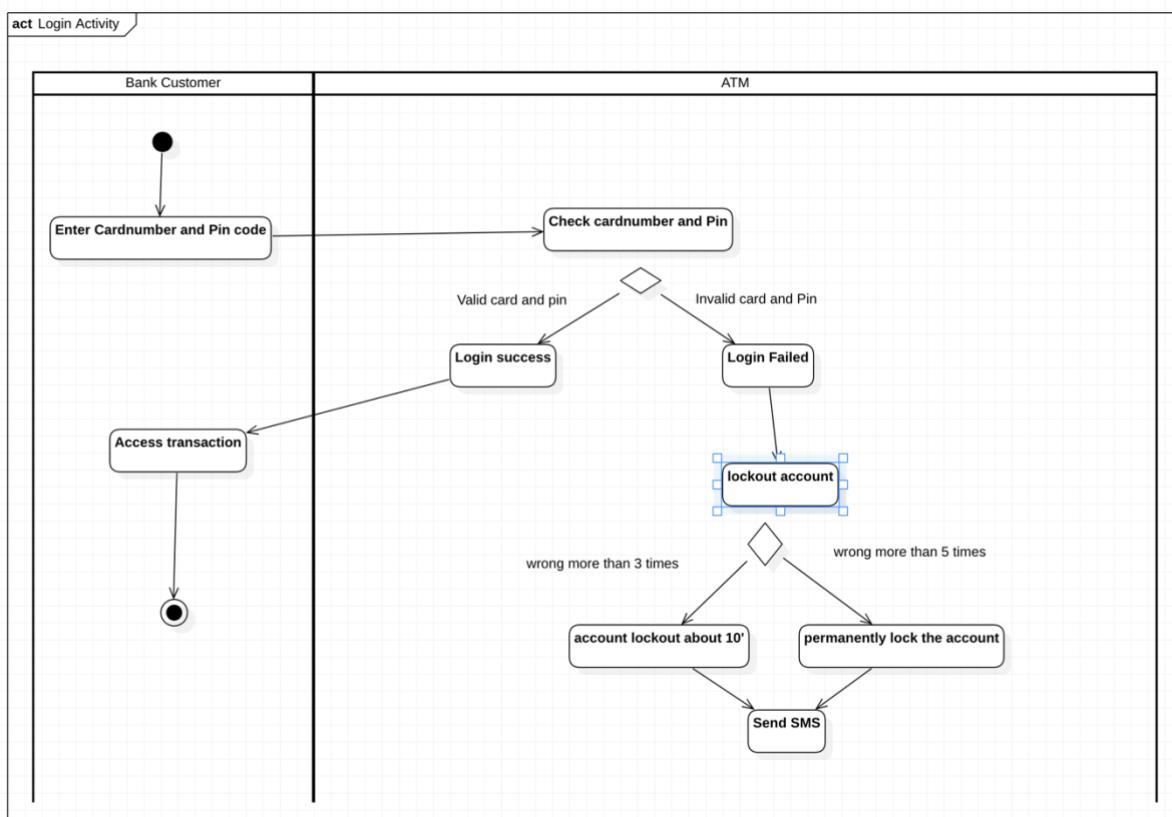


Hình 2.18: Sơ đồ UseCase phân rã nạp tiền

- Sau khi đăng nhập thành công trên ATM. Người dùng chọn chức năng gửi tiền vào tài khoản trên giao diện ATM. Hệ thống yêu cầu người dùng nhập số tiền cần nạp. Người dùng nhập số tiền cần nạp. Hệ thống yêu cầu người dùng xác nhận lại số tiền cần nạp. Người dùng xác nhận lại số tiền cần nạp. Hệ thống thực hiện giao dịch nạp tiền vào tài khoản của người dùng. Hệ thống cập nhật số dư tài khoản của người dùng

2.7 Mô hình Activity Diagram

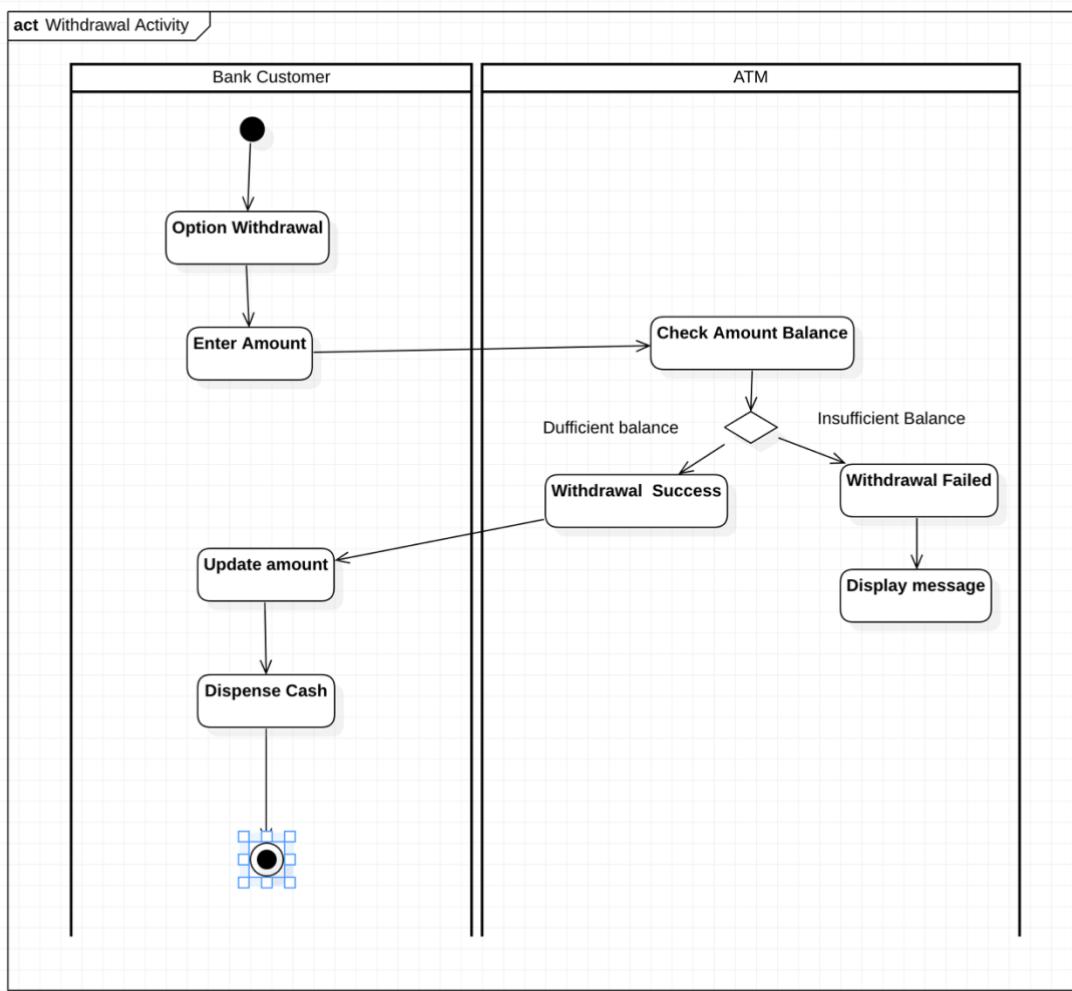
2.7.1 Mô hình Login Activity Diagram



Hình 2.19: Mô hình Login Activity Diagram

- Quá trình đăng nhập bắt đầu khi người dùng nhập số thẻ và mã PIN và kết thúc khi hệ thống truy cập thành công vào giao diện giao dịch. Sau đó, hệ thống sẽ kiểm tra thông tin này để xác thực khách hàng. Nếu thông tin là chính xác, hệ thống sẽ cho phép họ truy cập vào các chức năng của máy ATM như rút tiền, chuyển khoản, kiểm tra số dư và in sao kê. Nếu thông tin không chính xác, khách hàng sẽ không được phép truy cập vào hệ thống và họ phải thử lại hoặc tài khoản sẽ bị khoá trong 10 phút nếu nhập sai quá ba lần và bị hệ thống giam thẻ nếu nhập sai quá 5 lần, người dùng sẽ nhận được tin nhắn từ ngân hàng nếu nhập sai mã pin quá 3 hoặc 5 lần và họ phải liên hệ với ngân hàng để giải quyết vấn đề.

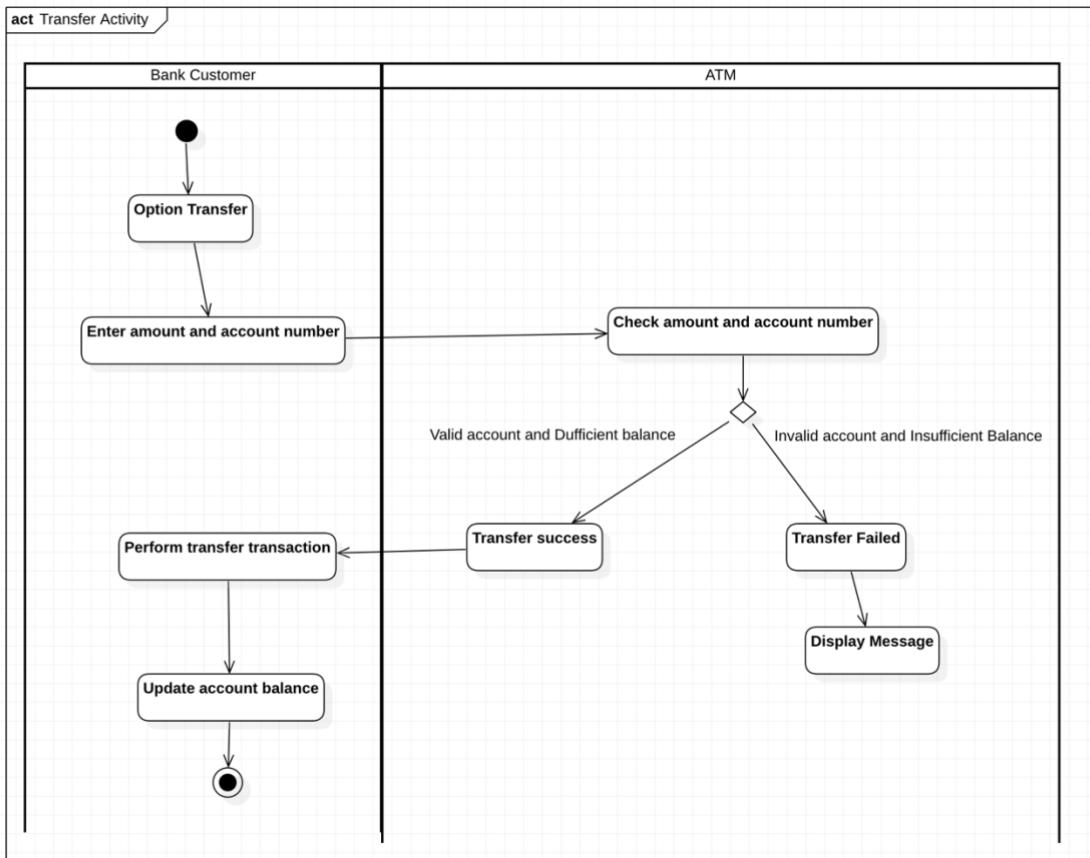
2.7.2 Mô hình Rút tiền Activity Diagram



Hình 2.20: Mô hình Rút tiền Activity Diagram

- Quá trình bắt đầu khi người dùng đăng nhập thành công vào atm. Người dùng chọn chức năng rút tiền trên màn hình ATM và nhập số tiền cần rút. Hệ thống kiểm tra số dư tài khoản để đảm bảo rằng tài khoản của người dùng có đủ tiền để thực hiện giao dịch rút tiền. Nếu số dư người dùng nhỏ hơn số tiền cần rút hệ thống sẽ gửi thông báo và yêu cầu người nhập số dư hợp lệ. Ngược lại nếu số dư lớn hơn số tiền cần rút thì hệ thống thực hiện giao dịch rút tiền bằng cách giảm số tiền tương ứng từ số dư tài khoản của người dùng. Hệ thống phát ra tiền mặt tương ứng với số tiền rút và thông báo số dư tài khoản mới cho người dùng. Người dùng lấy tiền và kết thúc giao dịch.

2.7.3 Mô hình Chuyển khoản Activity Diagram



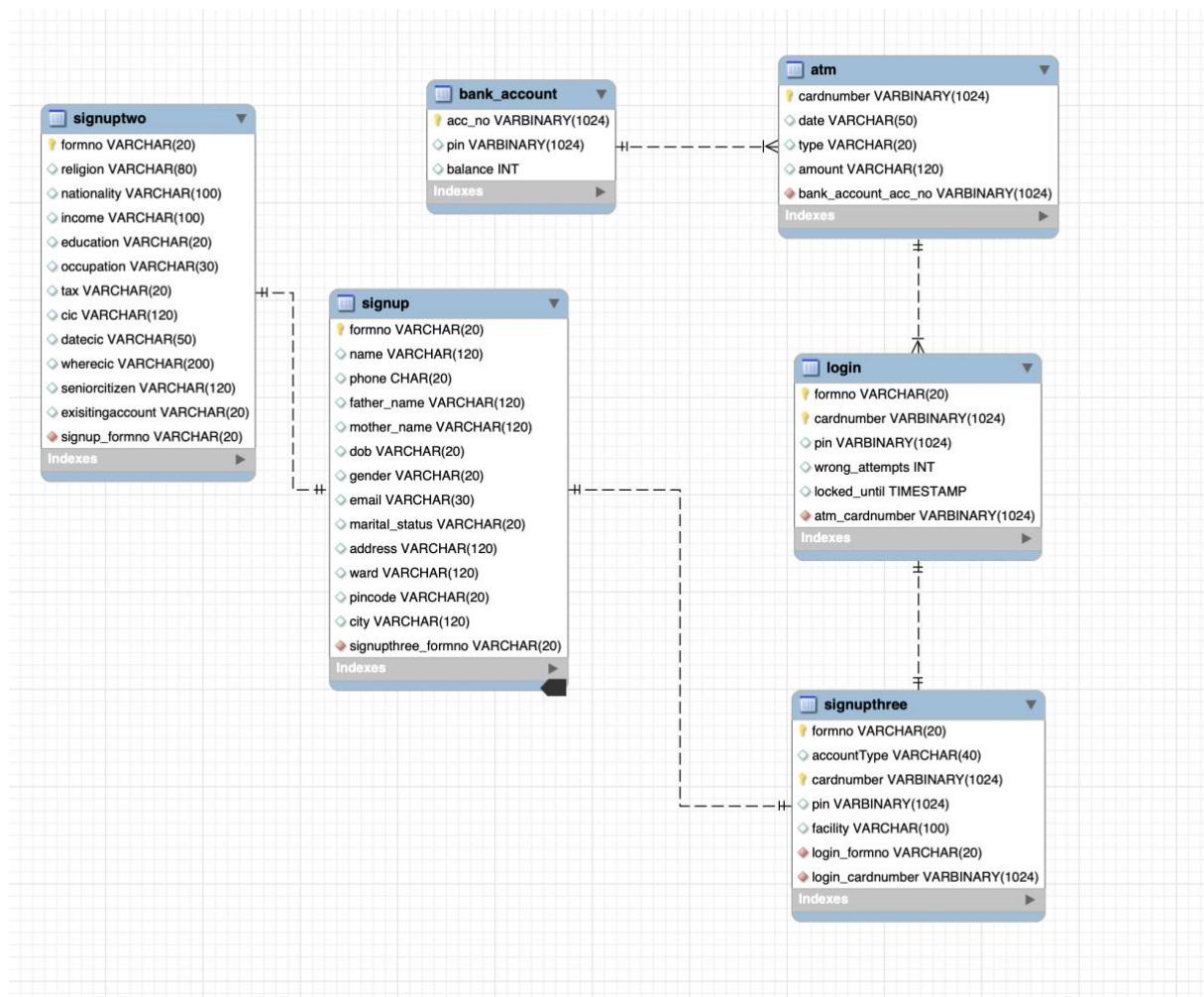
Hình 2.21: Mô hình chuyển khoản Activity Diagram

- Quá trình bắt đầu khi người dùng đăng nhập thành công vào atm. Người dùng chọn chức năng chuyển khoản trên màn hình ATM và người dùng cần nhập số tài khoản và số tiền cần chuyển. Sau đó, hệ thống sẽ kiểm tra số dư tài khoản của người dùng để xác định xem có đủ tiền để chuyển hay không. Nếu số dư đủ, hệ thống sẽ thực hiện giao dịch chuyển và cập nhật số dư tài khoản nguồn và tài khoản đích. Nếu số dư không đủ, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại số tiền cần chuyển hoặc nạp thêm tiền vào tài khoản. Nếu quá trình chuyển khoản thành công, người dùng sẽ được thông báo về kết quả giao dịch.

2.8 Phân tích thiết kế CSDL

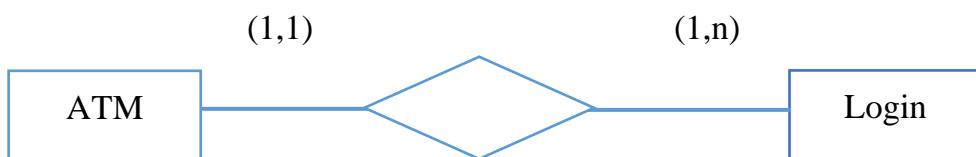
2.8.1 Xây dựng mô hình ER Diagram

Dựa theo quá trình khảo sát và kết quả phân tích thiết kế về các chức năng của hệ thống ATM tôi đã thiết kế ra mô hình ERD như sau:



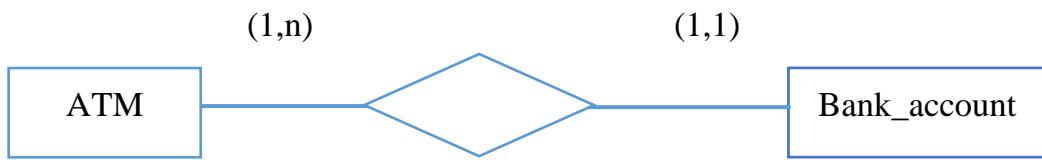
Hình 2.22: Mô hình ERD Diagram

2.8.2 Mô tả mối liên kết CSDL



Hình 2.23: Mô tả mối liên kết giữa bảng login và atm

Mô tả: Quan hệ giữa bảng Login và bảng ATM là quan hệ một-nhiều (one-to-many) vì một người dùng có thể thực hiện nhiều giao dịch tại nhiều máy ATM khác nhau, do đó mỗi bản ghi trong bảng Login có thể liên kết với nhiều bản ghi trong bảng ATM

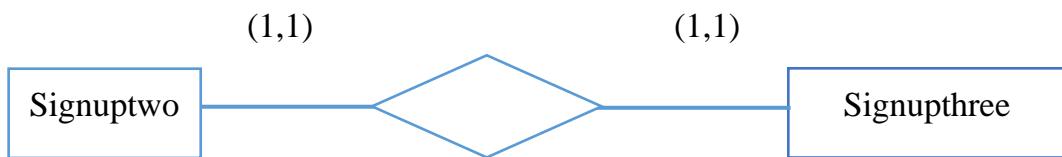


Hình 2.24: Mô tả mối liên kết giữa bảng atm và bank_account

Mô tả: Quan hệ giữa bảng atm và bảng bank_account là quan hệ một-nhiều (one-to-many), Mỗi tài khoản ngân hàng (bank_account) có thể liên kết với nhiều giao dịch rút tiền (atm), nhưng mỗi giao dịch rút tiền (atm) chỉ thuộc về một tài khoản ngân hàng (bank_account) duy nhất.



Hình 2.25: Mô tả mối liên kết giữa bảng signup và signuptwo



Hình 2.26: Mô tả mối liên kết giữa bảng signuptwo và signupthree

Mô tả: Quan hệ giữa ba bảng Signup, Signuptwo và Signupthree là một quan hệ một môt (one-to-one), vì mỗi bảng chỉ có duy nhất một bản ghi tương ứng với mỗi bản ghi trong bảng khác.

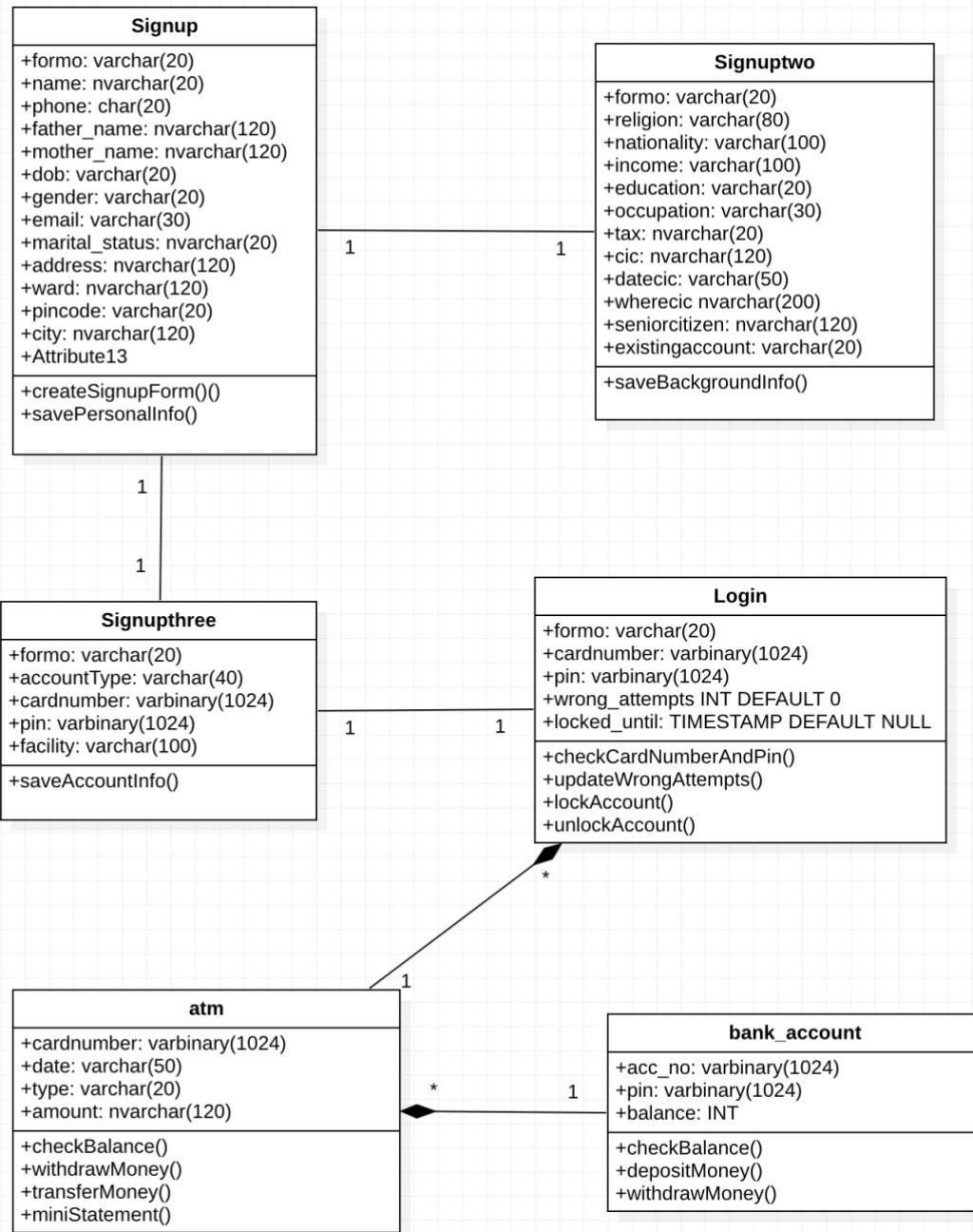
Cụ thể, bảng Signup lưu trữ thông tin cá nhân của người dùng, bảng Signuptwo lưu trữ các thông tin khác như tôn giáo, quốc tịch, thu nhập và bảng Signupthree lưu trữ thông tin về tài khoản của người dùng. Mỗi bản ghi trong bảng Signup có một bản ghi tương ứng trong bảng Signuptwo và Signupthree để cung cấp đầy đủ thông tin cho tài khoản của người dùng.



Hình 2.27: Mô tả mối liên kết giữa bảng signupthree và login

Mô tả: Quan hệ giữa bảng Signupthree và bảng Login là quan hệ một-một (one-to-one), vì mỗi tài khoản đăng nhập chỉ tương ứng với duy nhất một tài khoản được tạo trong bảng Signupthree. Ngược lại, mỗi tài khoản được tạo trong bảng Signupthree chỉ có thể được sử dụng để đăng nhập bằng một tài khoản duy nhất trong bảng Login. Quan hệ này đảm bảo tính chính xác và bảo mật trong việc xác thực người dùng.

2.9 Mô hình Class Diagram



Hình 2.28: Mô hình Class Diagram

Bảng 2.2: Bảng biểu diễn thuộc tính signup

Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc	Độ dài
formo	Mã đơn	Varchar	Khoá chính	20
signupthree_formo	Mã đơn số ba	Varchar	Khoá phụ	20
name	Họ và tên	Nvarchar		120
phone	Số điện thoại	Char		20
father_name	Họ tên cha	Nvarchar		120
mother_name	Họ tên mẹ	Nvarchar		120
dob	Ngày tháng năm sinh	Varchar		20
gender	Giới tính	Varchar		20
email	Địa chỉ email	Varchar		30
marital_status	Trạng thái hôn nhân	Nvarchar		20
address	Địa chỉ	Nvarchar		120
ward	Phường	Nvarchar		120
pincode	Mã số bưu chính	Varchar		20
city	Thành phố	Nvarchar		120

Bảng 2.3: Bảng biểu diễn thuộc tính Signuptwo

Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc	Độ dài
formo	Mã đơn	Varchar	Khoá chính	20
signuptwo_formo	Mã đơn số 2	Varchar	Khoá phụ	20
religion	Tôn giáo	Varchar		80
nationality	Quốc tịch	Varchar		100
income	Thu nhập	Varchar		100
education	Học vấn	Varchar		20
occupation	Nghề nghiệp	Varchar		30
tax	Mã số thuế	Nvarchar		20
cic	Căn cước công dân	Nvarchar		120
datecic	Ngày cấp căn cước	Varchar		50
wheretcic	Nơi cấp căn cước	Nvarchar		200
seniorcitizen	Người cao tuổi	Nvarchar		120
existingaccount	Tài khoản ngân hàng đã mở	Varchar		20

Bảng 2.4: Bảng biểu diễn thuộc tính Signupthree

Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc	Độ dài
formo	Mã đơn	Varchar	Khoá chính	20
login_formo	Mã đơn của bảng đăng nhập	Varchar	Khoá phụ	20
login_cardnumber	số thẻ của đăng nhập	Varbinary	Khoá phụ	1024
accountType	Loại tài khoản	Varchar		40
cardnumber	Số tài khoản thẻ	Varbinary		1024
pin	Mã pin	Varbinary		1024
facility	Dịch vụ	Varchar		100

Bảng 2.5: Bảng biểu diễn thuộc tính Login

Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc	Độ dài
formo	Mã đơn	Varchar	Khoá chính	20
atm_cardnumber	số thẻ của atm	Varbinary	Khoá phụ	1024
cardnumber	Số tài khoản thẻ	Varbinary		1024
pin	Mã pin	Varbinary		1024
wrong_attempts	Số lần nhập sai	Int		0
locked_until	Thời gian mở khoá	timestamp		Null

Bảng 2.6: Bảng biểu diễn thuộc tính ATM

Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc	Độ dài
cardnumber	Số tài khoản thẻ	Varbinary	Khoá chính	1024
bank_account_acc_no	Số thẻ của bank_account	Varbinary	Khoá phụ	1024
date	Ngày giao dịch	Varchar		50
type	Loại giao dịch	Varchar		20
amount	Số tiền giao dịch	Nvarchar		120

Bảng 2.7: Bảng biểu diễn thuộc tính Bank_account

Tên thuộc tính	Điễn giải	Kiểu dữ liệu	Ràng buộc	Độ dài
acc_no	Số tài khoản thẻ	Varbinary	Khoá chính	1024
pin	Mã pin	Varbinary		1024
balance	Số dư tài khoản	Int		

Chương 3: KẾT QUẢ THỰC NGHIỆM

3.1 Giao diện đăng nhập

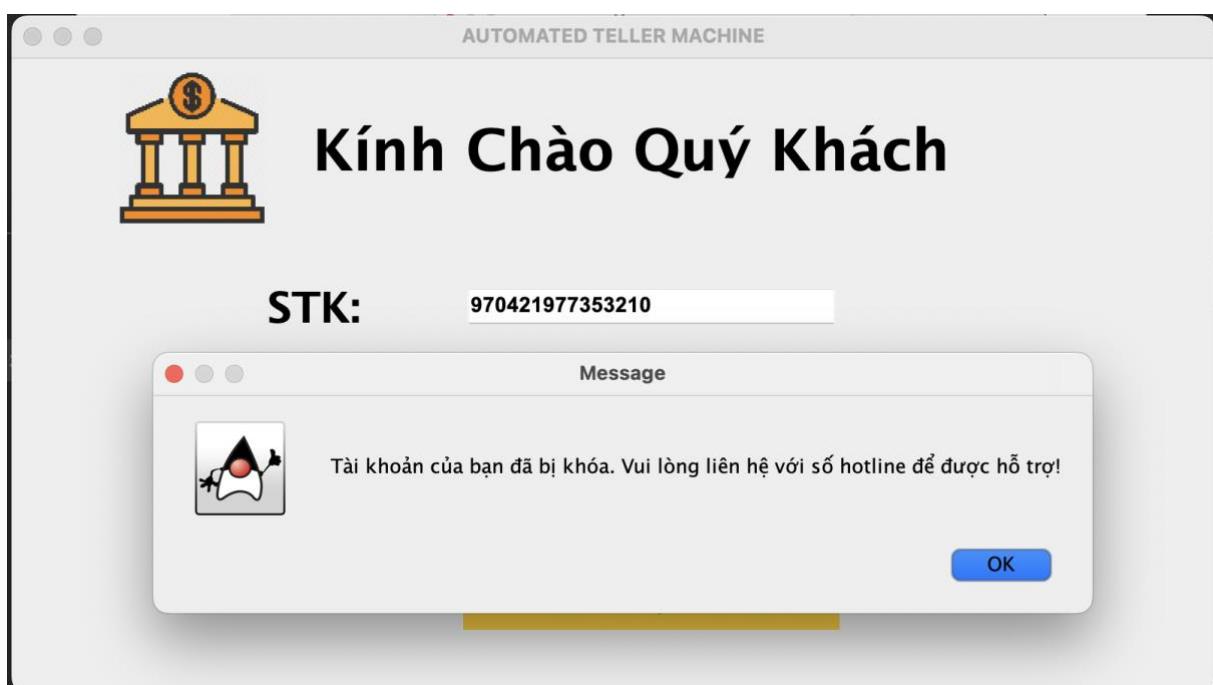


Hình 3.1: Giao diện đăng nhập vào ATM

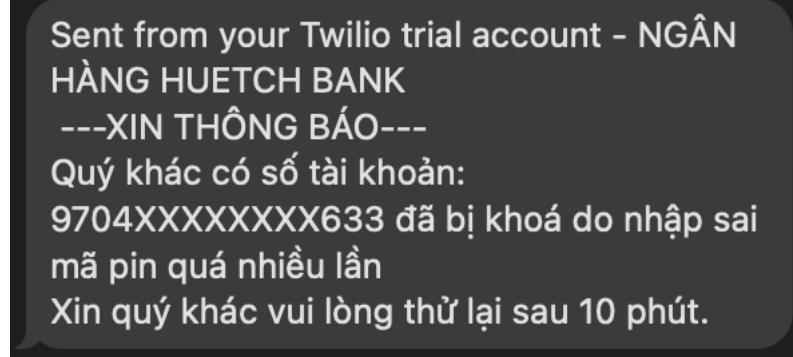
Giao diện đăng nhập là giao diện đầu tiên xuất hiện khi người dùng truy cập vào hệ thống ATM Simulator System. Giao diện này bao gồm một ô để nhập số tài khoản người dùng và mã pin. Người dùng cần phải nhập đúng thông tin của tài khoản đã đăng ký trước đó để được truy cập vào các chức năng của hệ thống.



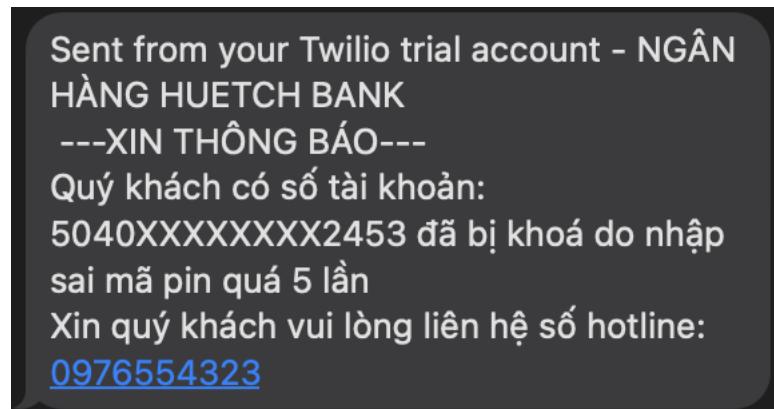
Hình 3.2: Thông báo nhập sai mã pin



Hình 3.3: Thông báo tài khoản bị khoá



Hình 3.0.4: Thông báo sms



Hình 3.5: Thông báo sms

Nếu thông tin nhập không chính xác, hệ thống sẽ thông báo lỗi và yêu cầu người dùng nhập lại. Nếu người dùng nhập sai quá nhiều hệ thống sẽ khoá tài khoản và gửi tin nhắn sms để thông báo với người.

3.2 Giao diện đăng ký

APPLICATION FORM NO. 686

Trang 1: Thông tin khách hàng

Họ tên: _____

SĐT: _____

Tên cha: _____

Tên mẹ: _____

Ngày sinh: _____

Giới tính: Nam Nữ

Email: _____

Hôn nhân: Đã kết hôn Chưa kết hôn Khách

Địa chỉ: _____

Phường: _____

Tỉnh/Thành Phố: _____

Post Code: _____

Tiếp theo

Hình 3.6: Mẫu điền thông tin số 1

NEW ACCOUNT APPLICATION FORM - PAGE 2

Trang 2: Thông tin bổ sung

Tôn giáo:

Quốc tịch:

Thu nhập:

Học vấn:

Nghề nghiệp:

Mã số thuế:
Vui lòng nhập đủ 10 số

CCCD:
Vui lòng nhập đủ 12 số

Ngày cấp:

Nơi cấp:

Người cao tuổi: Phải Không

TK Ngân Hàng: Có Chưa

Tiếp theo

Hình 3.7: Mẫu điền thông tin số 2

Trang 3: Chi tiết bổ sung

Loại tài khoản:

Tài khoản tiết kiệm Tài khoản thanh toán
 Tài khoản tín dụng Tài khoản cho vay

STK: XXXX-XXXX-XXXX-4134
Vui lòng nhập đủ 16 số

PIN: XXXXXX
Vui lòng nhập đủ 6 số

Dịch vụ cần thiết:

ATM CARD Internet Banking
 Mobile Banking Email & sms banking
 Cheque Book E-Statement

Tôi cam kết hoàn toàn chịu trách nhiệm toàn bộ lời khai trên

Xác nhận Huỷ

Hình 3.8: Mẫu điền thông tin số 3

Giao diện đăng ký cho phép người dùng kê khai thông tin để mở tài khoản ngân hàng mới để sử dụng trong ATM. Giao diện này bao gồm các thông tin cần thiết như tên người dùng, địa chỉ email, số điện thoại, địa chỉ nhà, căn cước công dân, quốc tịch, dịch vụ ngân hàng... Người dùng cần phải nhập đúng và đầy đủ thông tin yêu cầu để hoàn tất quá trình đăng ký.

3.3 Giao diện tổng quát – atm



Hình 3.9: Giao diện ATM

Giao diện tổng quát của hệ thống ATM cho phép người dùng truy cập vào tất cả các chức năng của hệ thống. Giao diện này bao gồm các phím chức năng để rút tiền, gửi tiền, chuyển tiền, kiểm tra số dư, in sao kê và đổi mã pin. Người dùng có thể chọn chức năng mà mình muốn sử dụng bằng cách nhấn vào các phím tương ứng.

3.4 Giao diện rút tiền

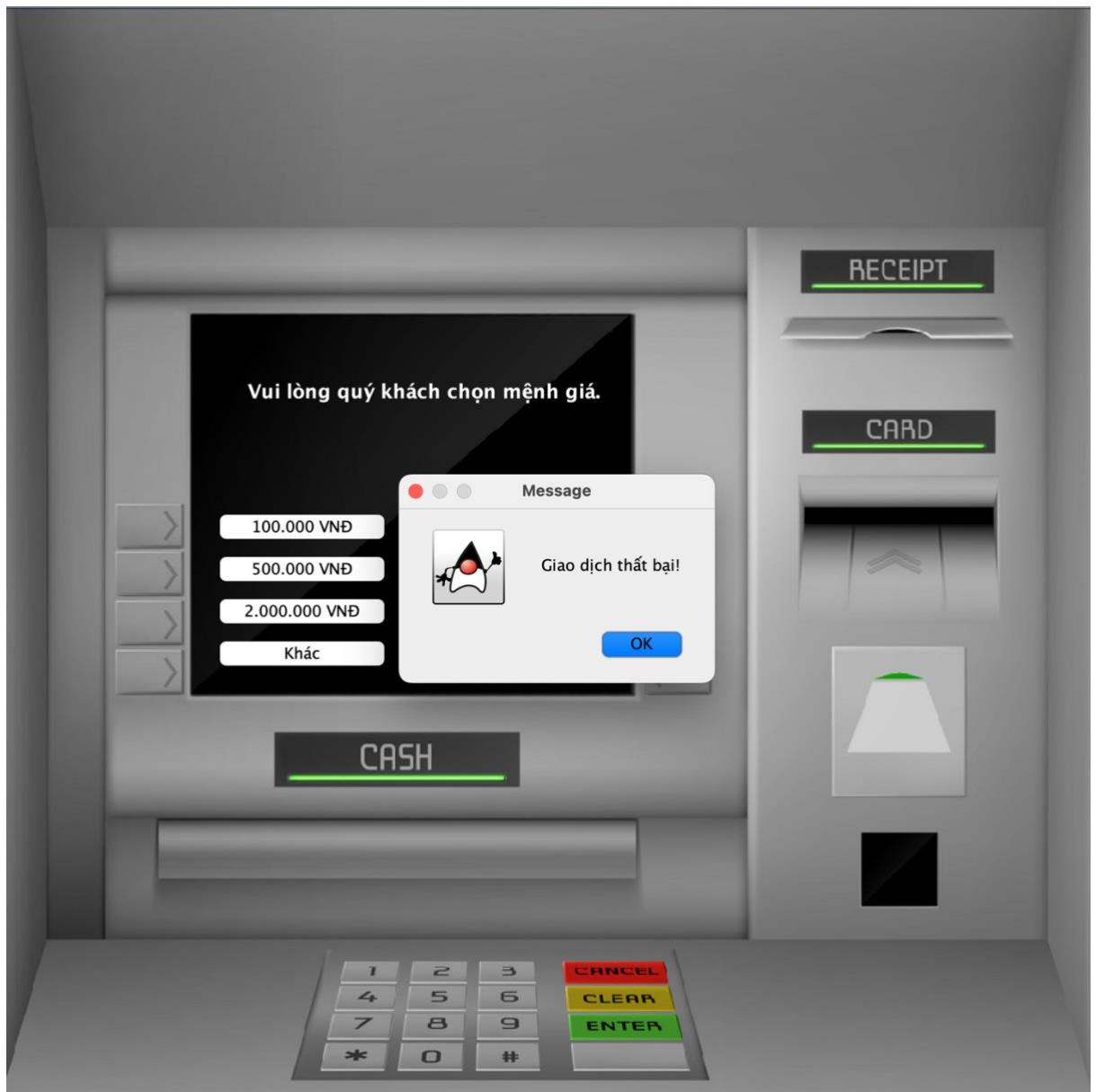


Hình 3.10: Giao diện rút tiền nhanh

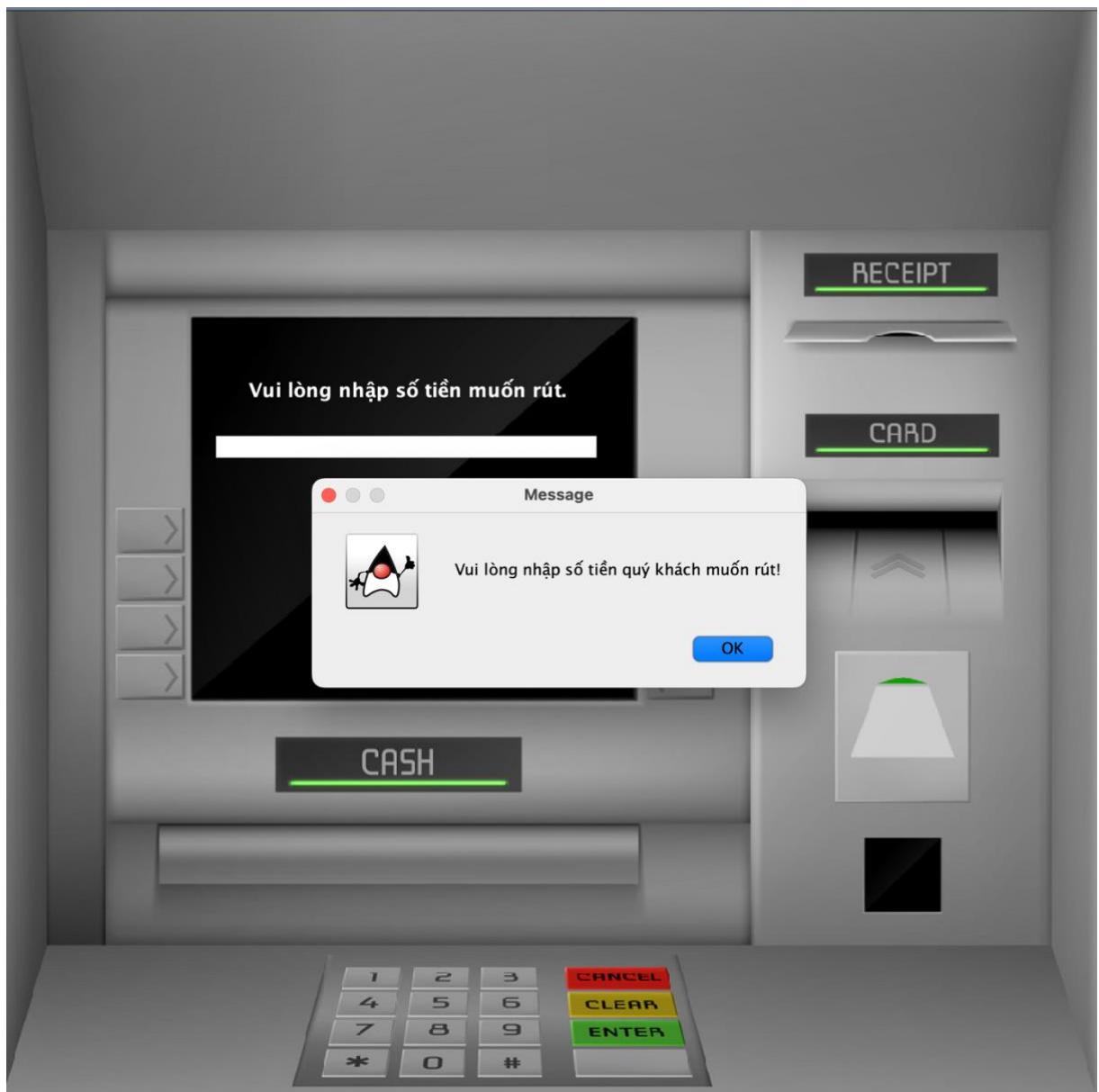
Giao diện rút tiền của hệ thống ATM được thiết kế với giao diện đơn giản và dễ sử dụng. Khi khách hàng muốn thực hiện giao dịch rút tiền, họ sẽ chọn "Rút tiền" trên giao diện tổng quát của ATM.



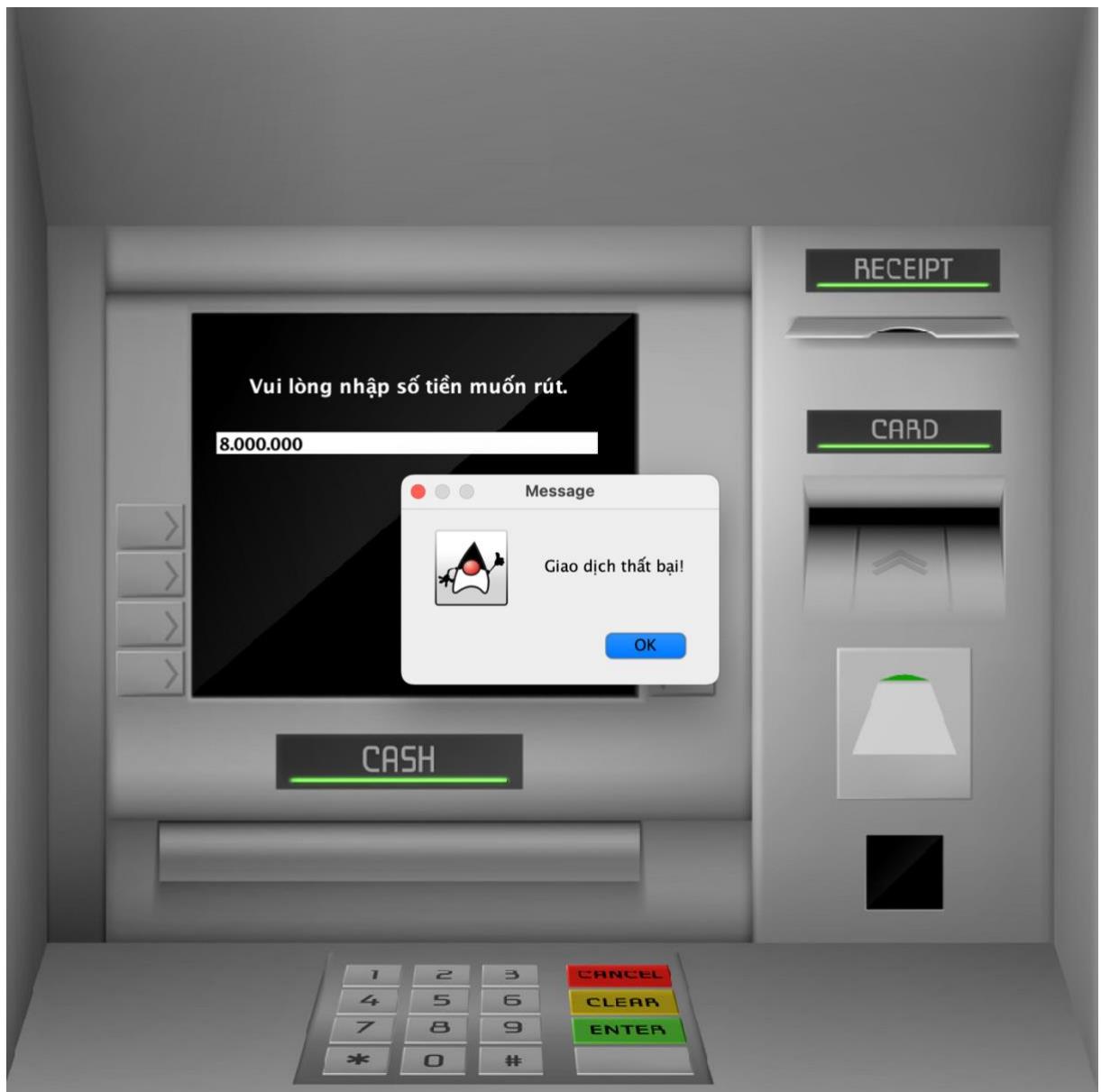
Hình 3.11: Giao diện rút tiền



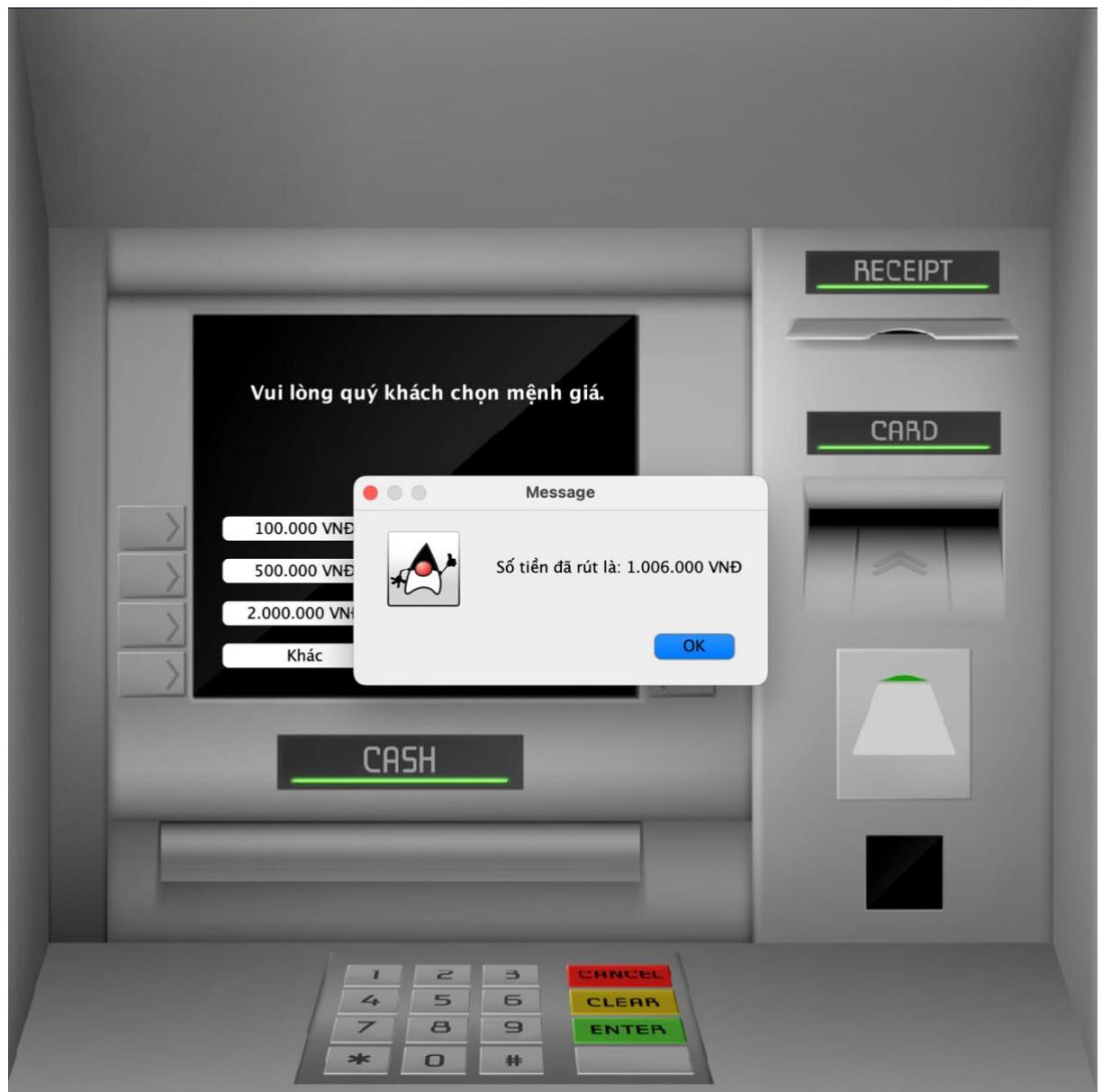
Hình 3.12: Thông báo số dư tài khoản không đủ



Hình 3.13: : Kiểm tra thông tin nhập



Hình 3.14: Kiểm tra số dư tài khoản



Hình 3.15: Thông báo rút tiền thành công



Hình 3.16: Thông báo rút tiền thành công

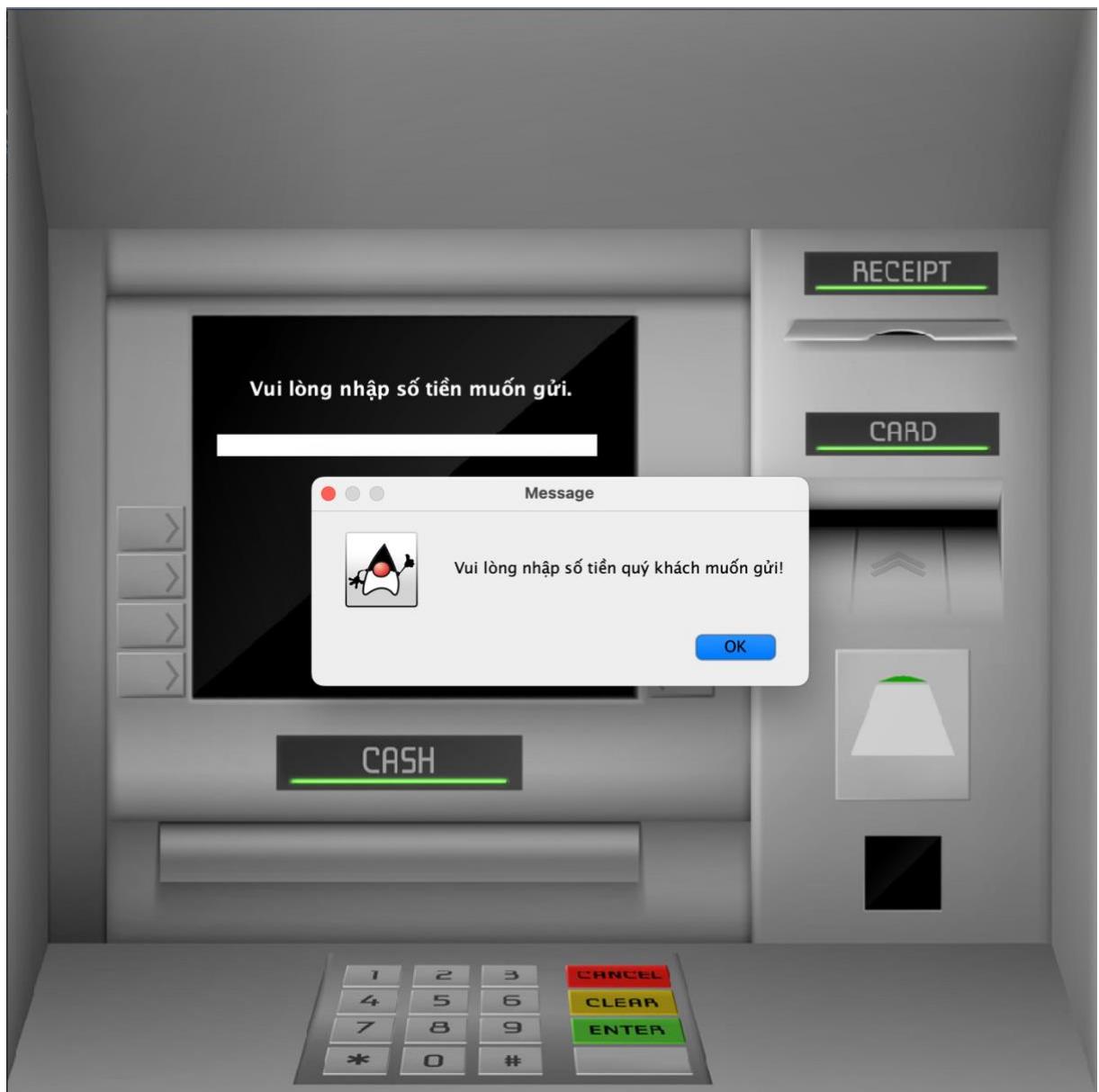
Sau đó, khách hàng sẽ nhập số tiền cần rút hoặc chọn mệnh giá của tiền vào các trường thông tin tương ứng trên giao diện.

Sau khi nhập thông tin và xác nhận giao dịch, hệ thống sẽ kiểm tra số dư tài khoản của khách hàng. Nếu số dư đủ để thực hiện giao dịch rút tiền, tiền sẽ được rút ra và người dùng sẽ được cung cấp hóa đơn và tiền mặt tương ứng với số tiền rút. Nếu không đủ số dư, hệ thống sẽ thông báo cho người dùng biết và yêu cầu họ nhập số tiền nhỏ hơn hoặc nạp tiền vào tài khoản để thực hiện giao dịch. Khách hàng khi rút tiền ATM sẽ phải trả thêm phí dịch vụ. Phí dịch vụ tuỳ theo mỗi ngân hàng quy định. Phí dịch vụ ở đây là 6 ngàn cho mỗi lần rút.

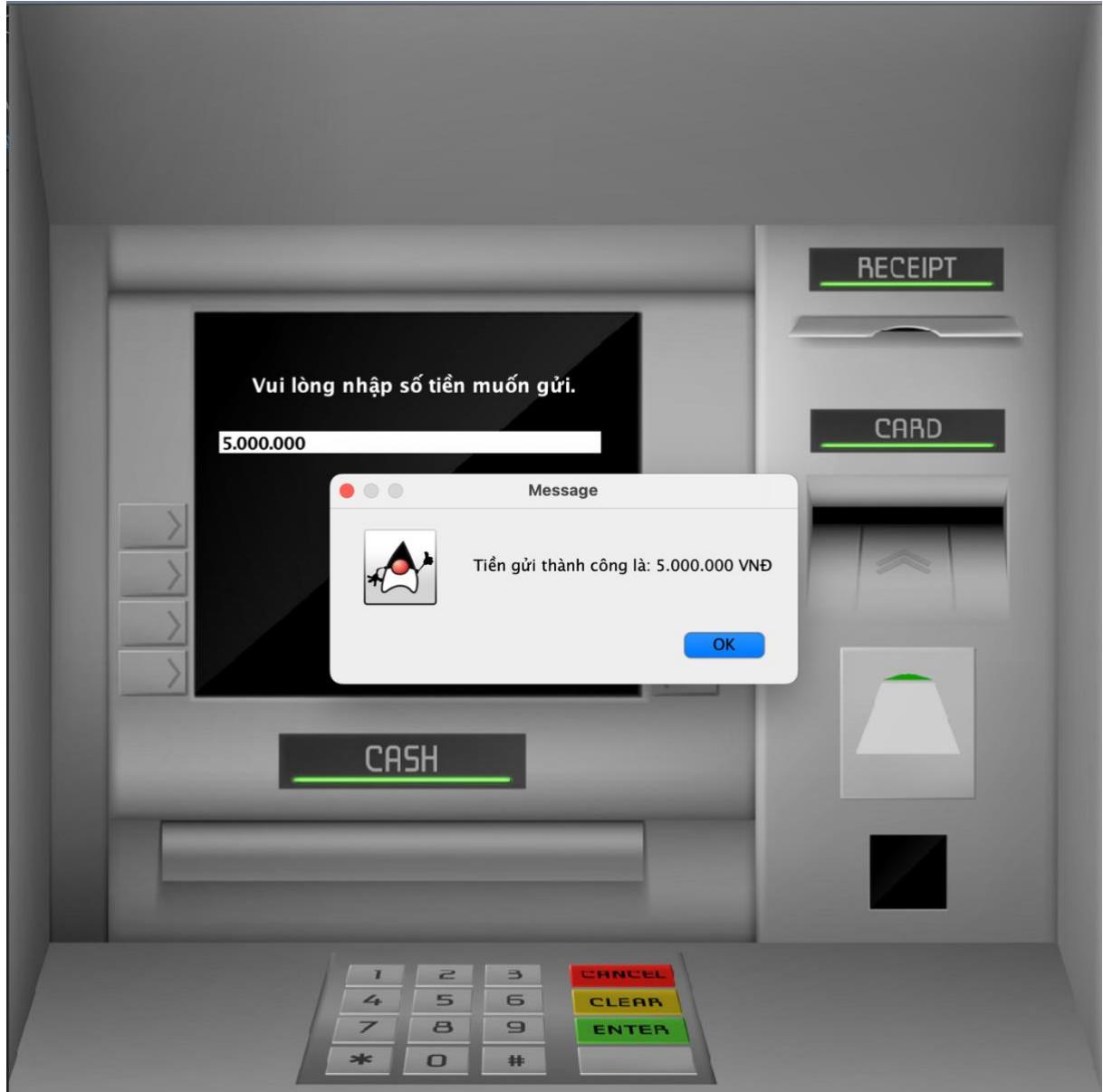
3.5 Giao diện gửi tiền



Hình 3.17: Giao diện gửi tiền vào tài khoản



Hình 3.18: Kiểm tra thông tin nhập



Hình 3.19: Thông báo nạp tiền thành công

Giao diện gửi tiền được thiết kế đơn giản và dễ sử dụng cho người dùng. Giao diện này bao gồm các trường thông tin cần nhập như số tiền cần gửi.

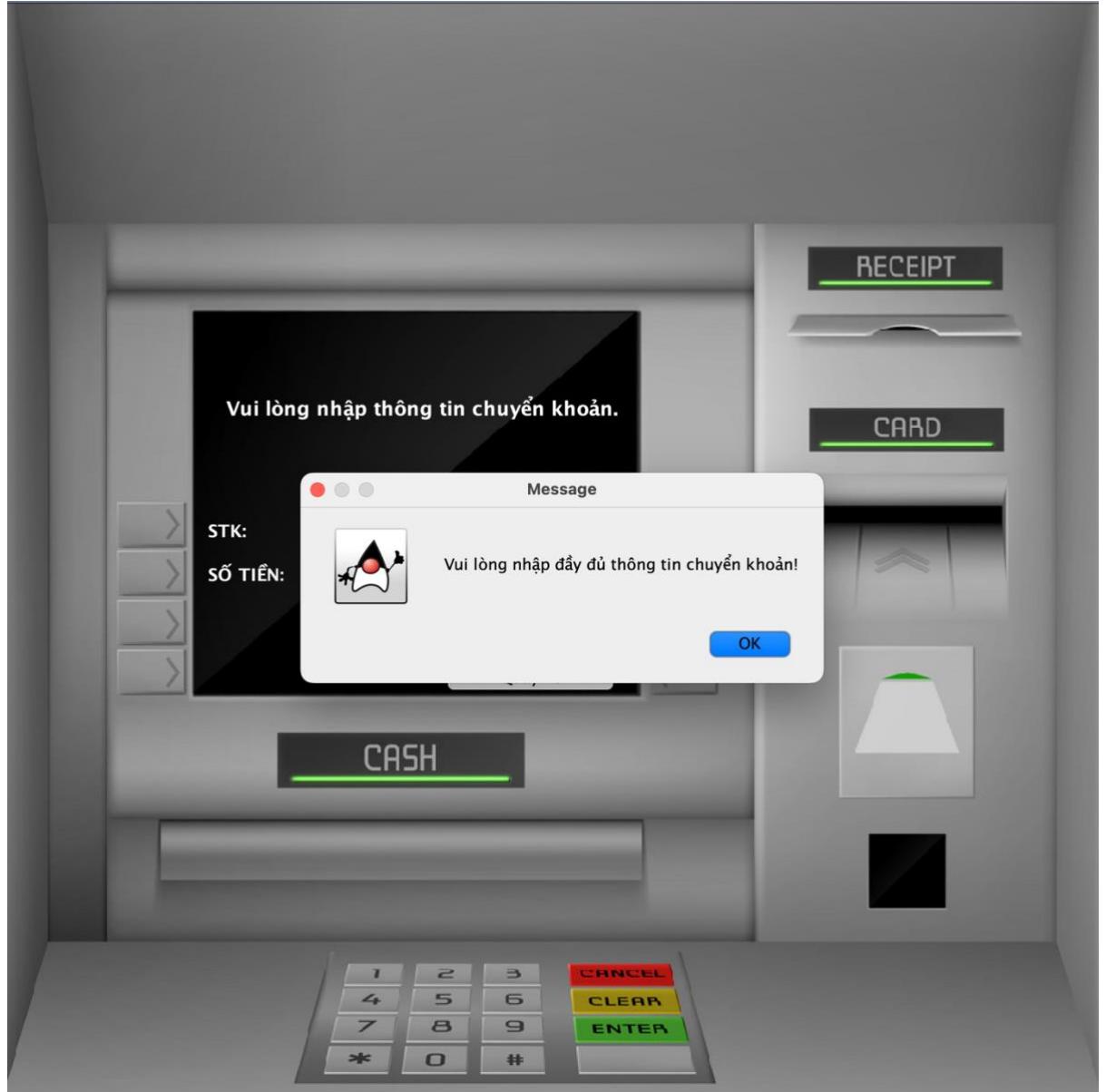
Khi người dùng nhập số tiền cần gửi, hệ thống sẽ tự động kiểm tra số dư trong tài khoản của người dùng để đảm bảo rằng số tiền gửi không vượt quá số dư hiện có. Nếu số tiền gửi vượt quá số dư, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại số tiền hợp lệ.

3.6 Giao diện chuyển tiền

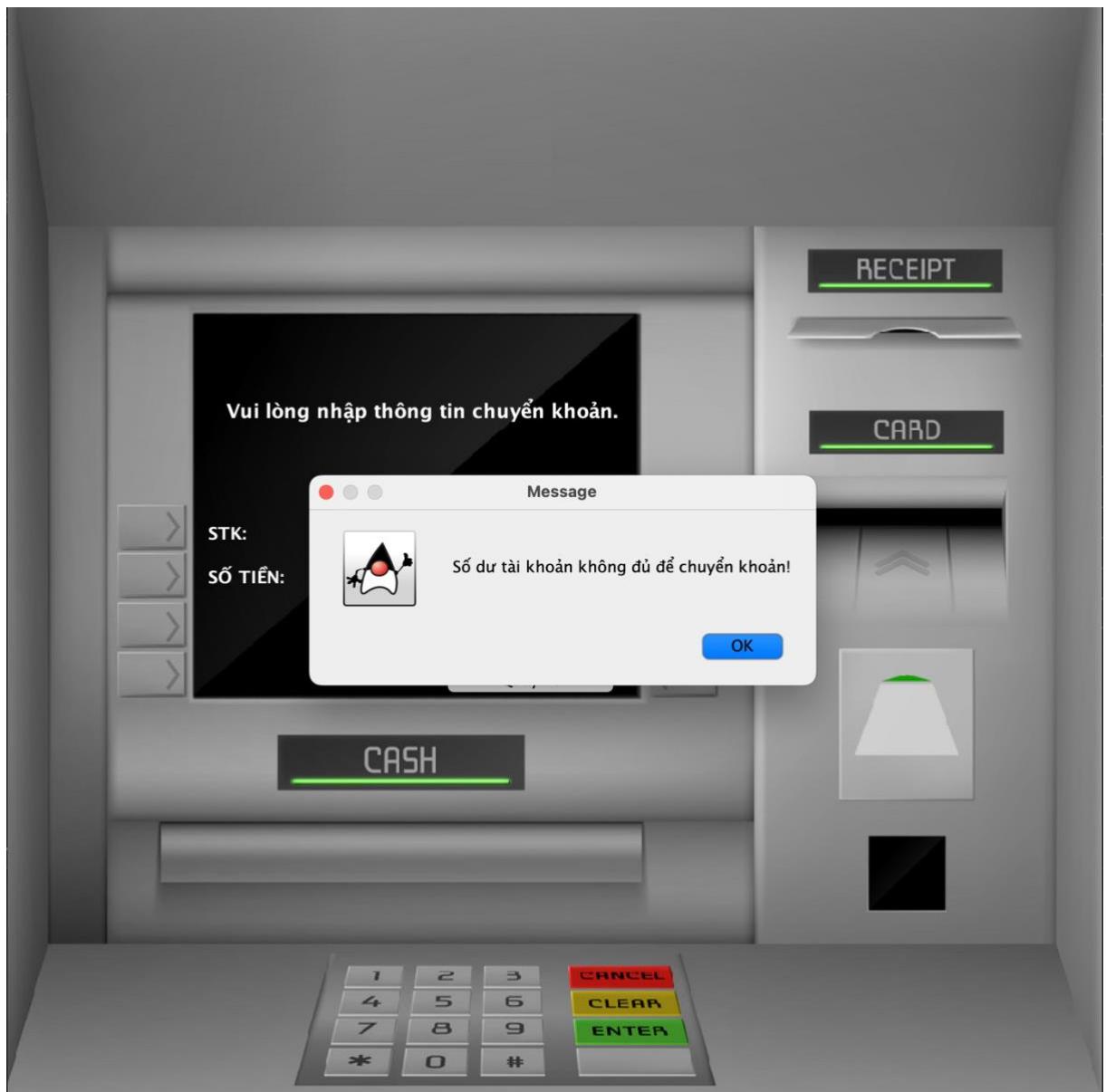


Hình 3.20: Giao diện chuyển tiền

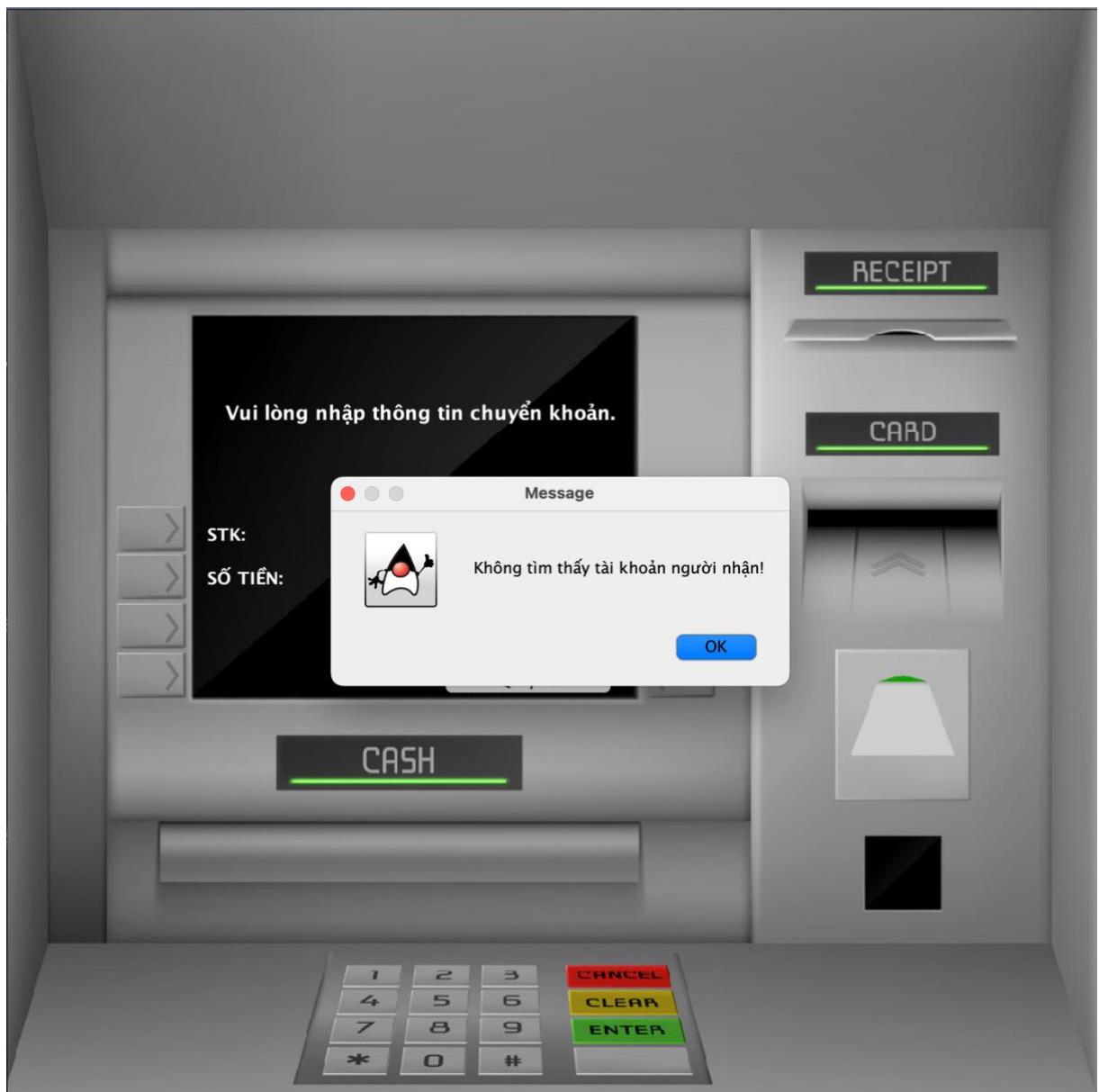
Giao diện chuyển tiền cho phép người dùng chuyển tiền từ tài khoản của mình sang tài khoản khác. Giao diện này bao gồm các thông tin cần thiết như số tiền cần chuyển, số tài khoản người nhận. Người dùng cần phải nhập đúng và đầy đủ thông tin yêu cầu để hoàn tất giao dịch chuyển tiền.



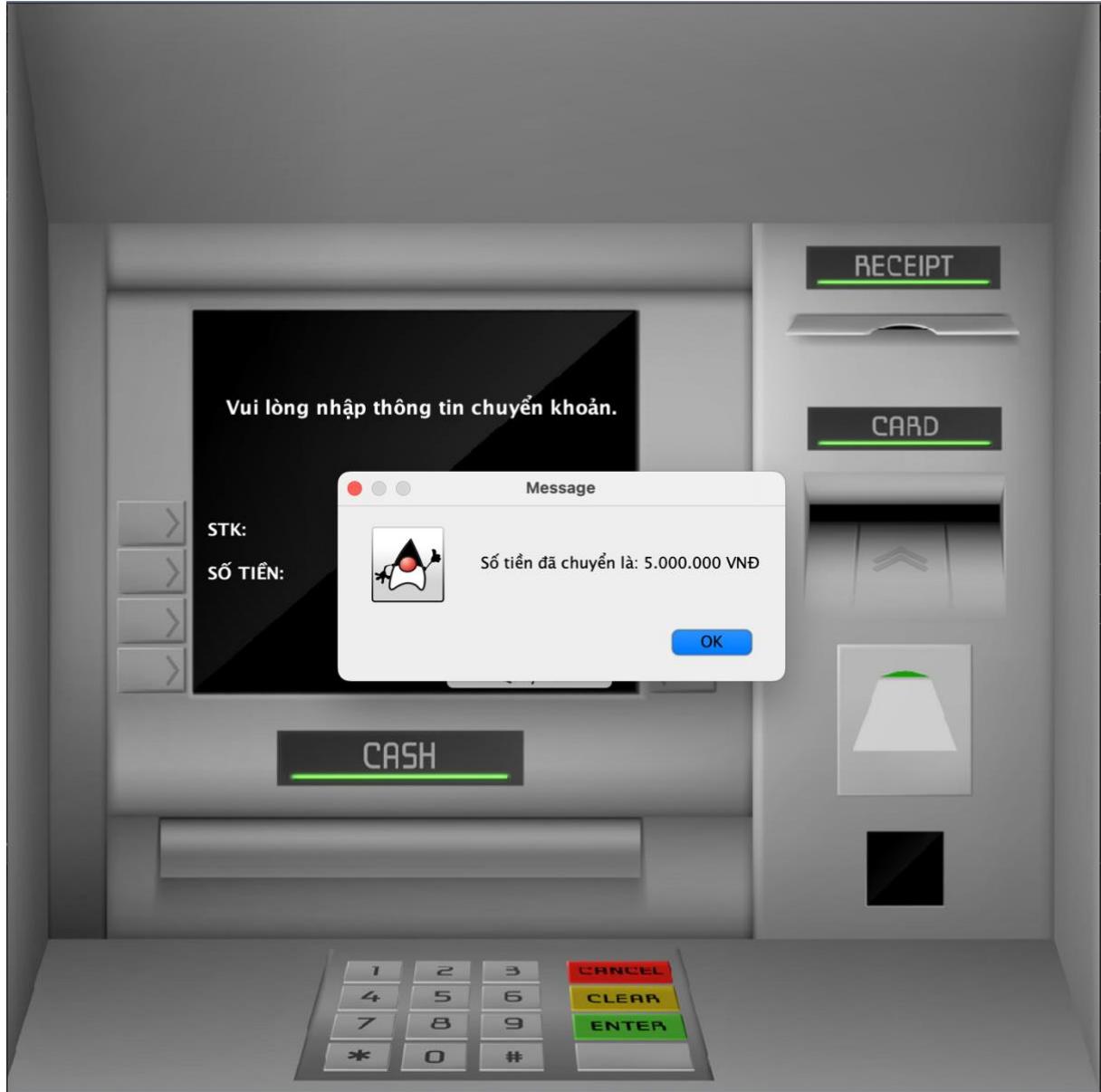
Hình 3.21: Kiểm tra thông tin nhập



Hình 3.22: Kiểm tra số dư tài khoản



Hình 3.23: Kiểm tra tài khoản



Hình 3.24: Thông báo chuyển tiền thành công

Giao diện được thiết kế đơn giản và dễ sử dụng, với các ô nhập liệu rõ ràng và các nút chức năng dễ nhận biết. Khi người dùng nhập đúng thông tin và nhấn nút "Chuyển tiền", hệ thống sẽ tiến hành xác thực thông tin và thực hiện giao dịch chuyển tiền.

Nếu giao dịch thành công, hệ thống sẽ thông báo cho người dùng biết và cập nhật lại số dư tài khoản. Ngược lại, nếu giao dịch thất bại, hệ thống sẽ thông báo lỗi và yêu cầu người dùng thực hiện lại giao dịch với thông tin chính xác.

3.7 Giao diện kiểm tra số dư tài khoản



Hình 3.25: Giao diện kiểm tra số dư

Giao diện kiểm tra số dư tài khoản cho phép người dùng xem số dư hiện tại của tài khoản của mình. Hệ thống sẽ hiển thị số dư hiện tại của tài khoản đó trên giao diện, giúp người dùng có thể dễ dàng theo dõi tình hình tài chính của mình.

3.8 Giao diện in sao kê

Hutech Bank

Card number: 9704XXXXXXX747

2023-06-08	Gửi tiền	1000000
2023-06-08	Gửi tiền	10000000
Thu Jun 08 21:19:32 ICT 2023	Rút tiền	500000
Thu Jun 08 21:20:16 ICT 2023	Rút tiền	800000
Thu Jun 08 21:20:26 ICT 2023	Rút tiền	100000
Thu Jun 08 21:24:14 ICT 2023	Chuyển khoản	1000000
Fri Jun 09 19:42:19 ICT 2023	Rút tiền	8000000
2023-06-09	Gửi tiền	10000000
Fri Jun 09 19:48:55 ICT 2023	Rút tiền	1000000
Fri Jun 09 19:51:27 ICT 2023	Rút tiền	2000000
2023-06-09	Gửi tiền	5000000
Fri Jun 09 19:57:02 ICT 2023	Chuyển khoản	1000000
Fri Jun 09 20:04:54 ICT 2023	Chuyển khoản	5000000

Số dư tài khoản là: 6.564.000 VNĐ

Số dư bằng chữ: sáu triệu năm trăm sáu mươi bốn nghìn đồng

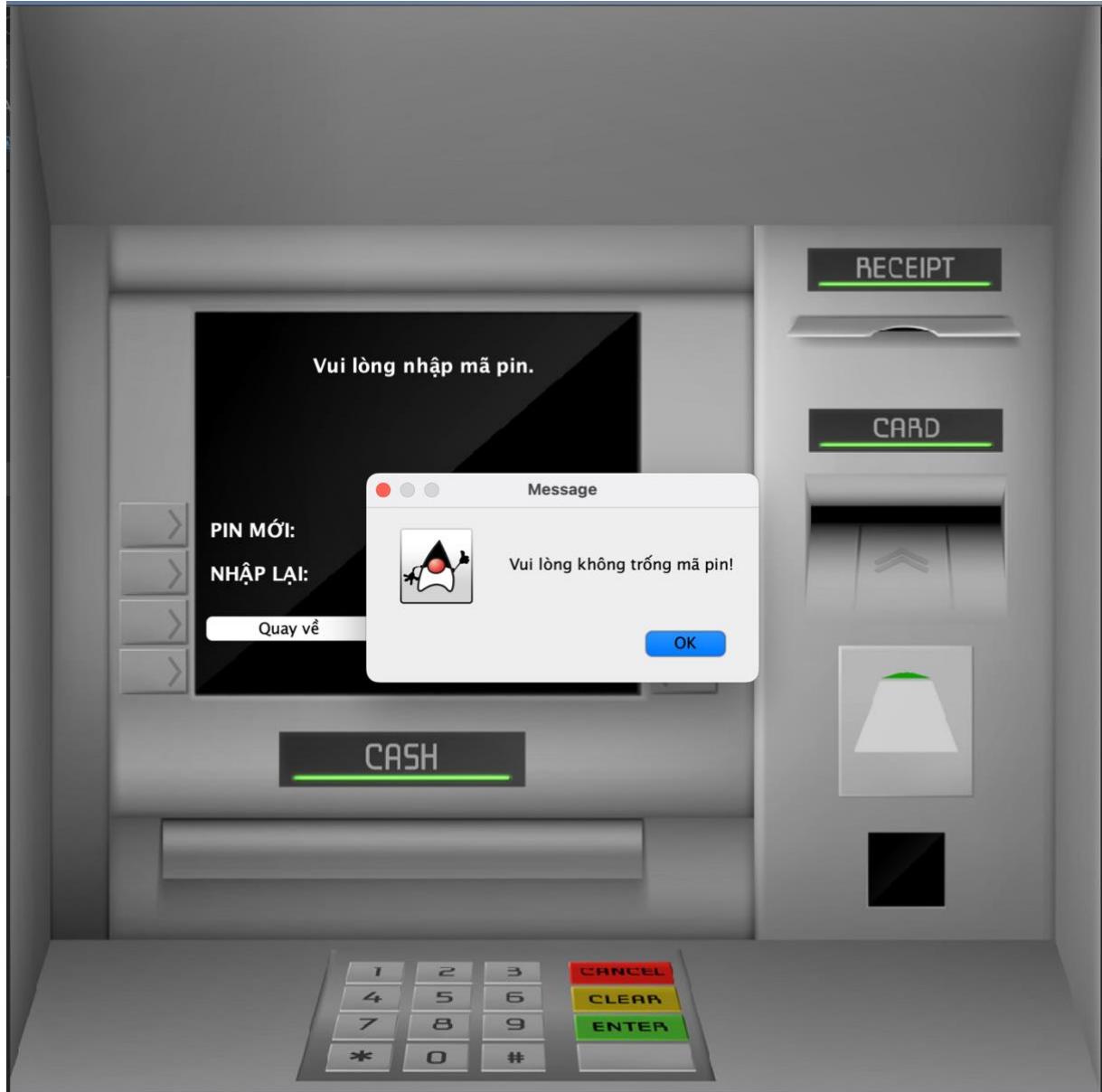
Hình 3.26: Xuất phiếu In sao kê

Giao diện in sao kê cho phép người dùng in thông tin chi tiết về các giao dịch đã thực hiện trên tài khoản của mình. Giao diện này hiển thị các thông tin như ngày thực hiện giao dịch, loại giao dịch (rút tiền, gửi tiền, chuyển tiền), số tiền giao dịch và số dư hiện tại của tài khoản. Người dùng có thể in các thông tin này ra để lưu trữ hoặc kiểm tra lại lịch sử giao dịch trên tài khoản của mình.

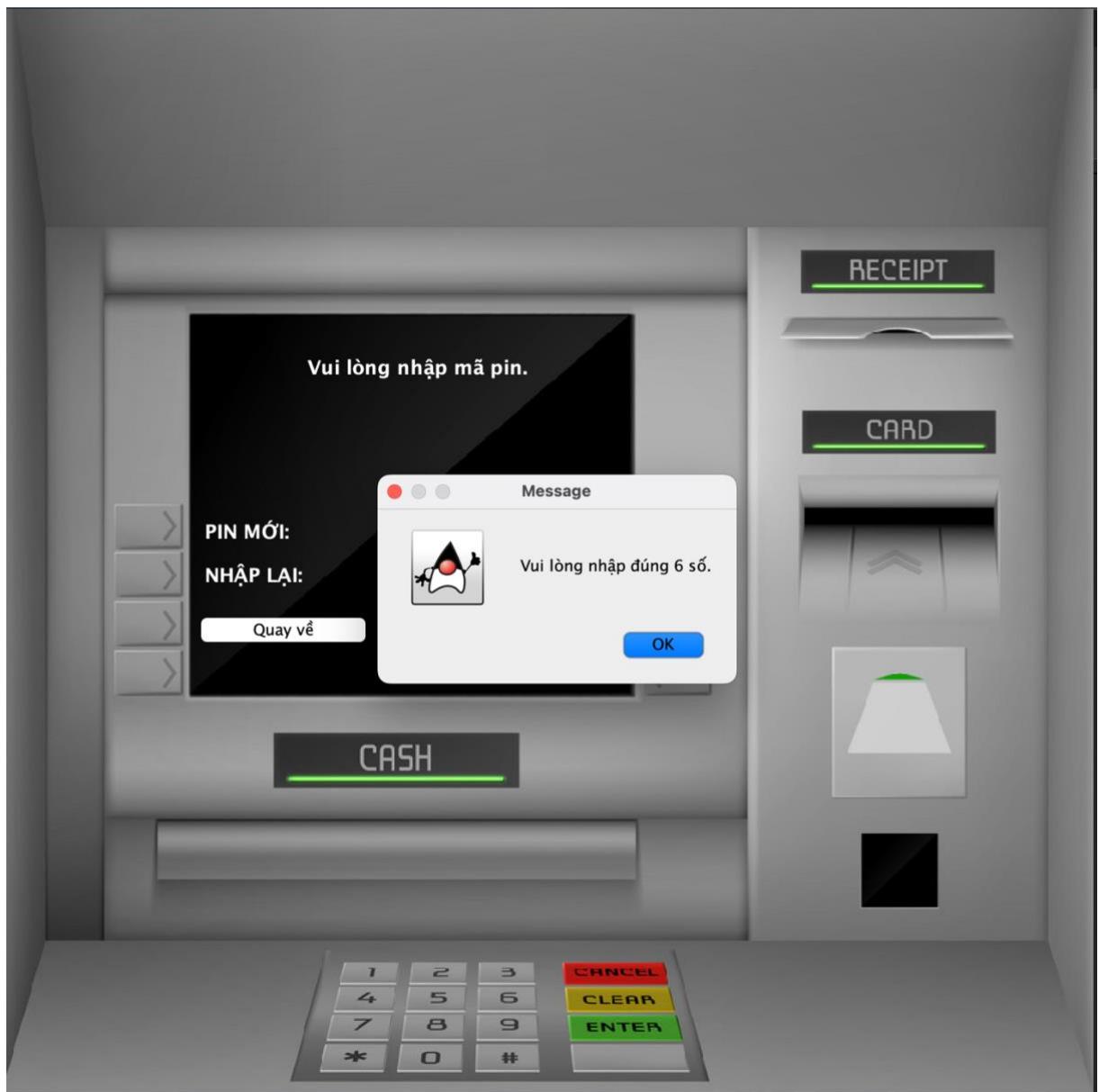
3.9 Giao diện đổi mã pin



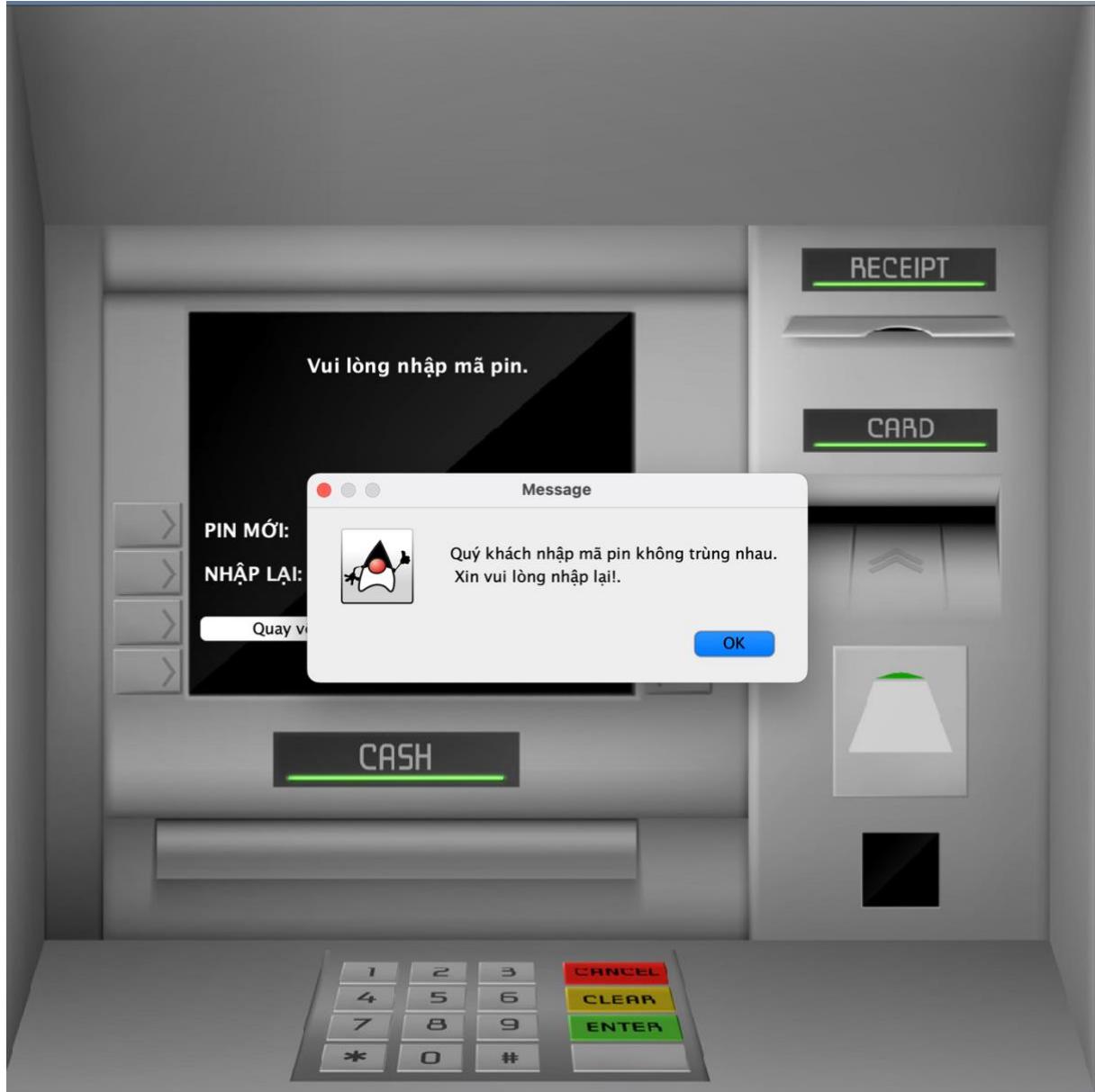
Hình 3.27: Giao diện đổi mã pin



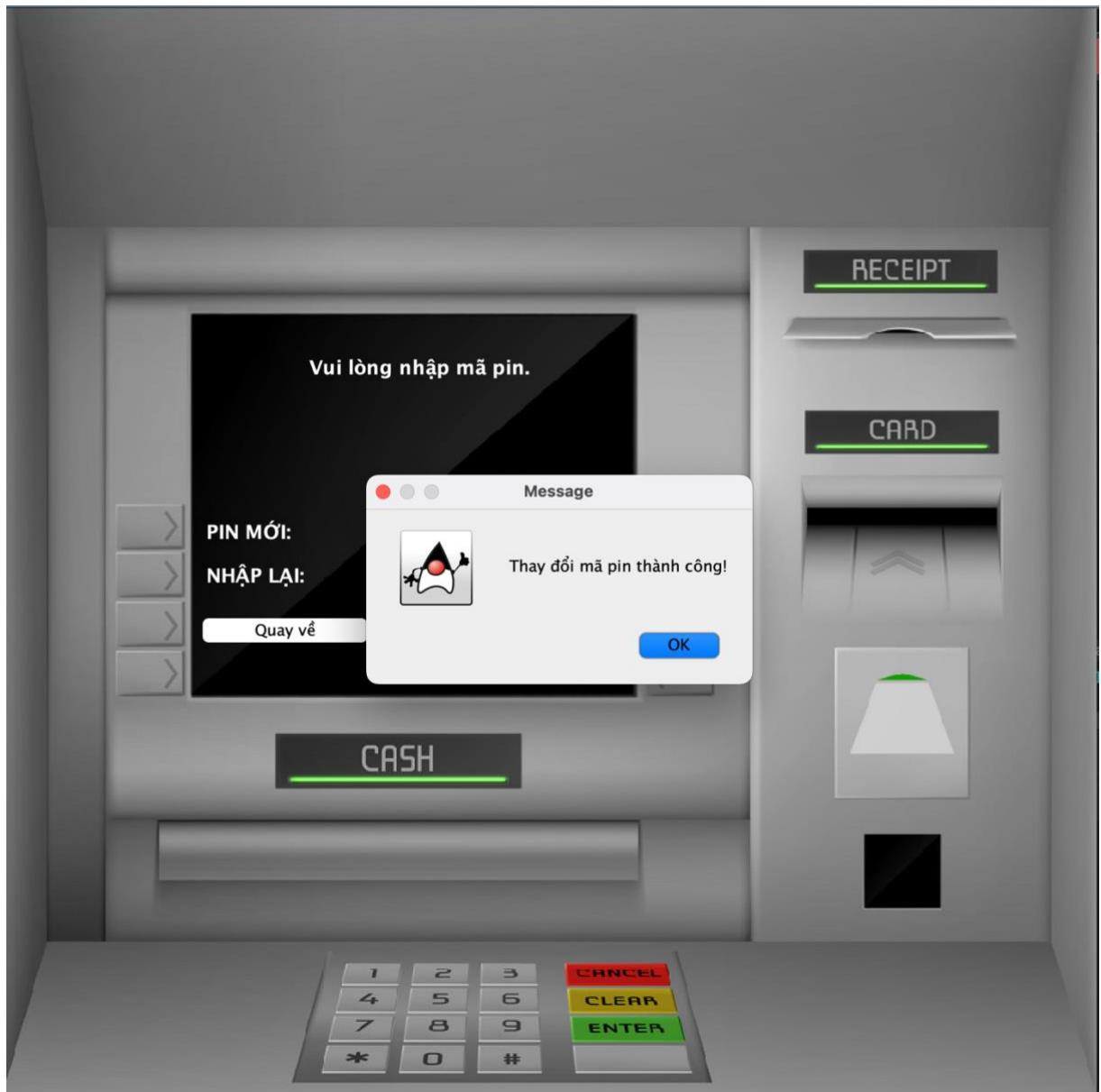
Hình 3.28: Kiểm tra thông tin nhập



Hình 3.29: : Kiểm tra điều kiện nhập



Hình 3.30: Kiểm tra điều kiện nhập



Hình 3.31: Thông báo thay đổi mã pin thành công

Giao diện đổi mã PIN cho phép người dùng thay đổi mã PIN của mình. Giao diện này bao gồm các ô nhập để người dùng nhập mã PIN cũ và mã PIN mới. Sau khi nhập thông tin, người dùng cần nhấn nút "Xác nhận" để hoàn tất việc thay đổi mã PIN. Nếu thông tin nhập vào không chính xác, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại. Nếu quá trình thay đổi mã PIN thành công, hệ thống sẽ hiển thị thông báo thành công và yêu cầu người dùng đăng nhập lại bằng mã PIN mới để sử dụng các dịch vụ của hệ thống.

Chương 4: KẾT LUẬN VÀ KIẾN NGHỊ

4.1. Kết quả đạt được

Sau quá trình phát triển, hệ thống ATM Simulator System đã đạt được các kết quả như sau:

1. Hoàn thành các chức năng cơ bản của một hệ thống ATM như đăng nhập, đăng ký, rút tiền, gửi tiền, chuyển tiền, kiểm tra số dư, in sao kê và đổi mã pin.
2. Áp dụng thành công các công nghệ phổ biến như Java Swing, MySQL Server, mã hóa RSA và SHA-512, hệ thống quản lý SMS Twilio để phát triển hệ thống.
3. Thiết kế giao diện người dùng đơn giản và dễ sử dụng, giúp người dùng dễ dàng sử dụng các tính năng của hệ thống.
4. Đảm bảo tính an toàn, tính năng và hiệu suất cao của hệ thống.

4.2 Đánh giá sản phẩm

Sản phẩm đã đáp ứng được các yêu cầu cơ bản của một hệ thống ATM và hoạt động ổn định. Giao diện đơn giản, thân thiện với người dùng, tuy nhiên vẫn còn một số tính năng cần phát triển và cải thiện để tăng tính năng, tiện ích và độ bảo mật của hệ thống.

4.3 Hướng phát triển và mở rộng đề tài

Để cải thiện sản phẩm, có thể thực hiện các hướng phát triển và mở rộng đề tài như sau:

1. Cải thiện tính bảo mật của hệ thống bằng cách sử dụng các công nghệ mã hóa tiên tiến hơn, chẳng hạn như mã hoá AES.
2. Tăng cường tính năng và tiện ích của hệ thống, bằng cách bổ sung thêm các chức năng như lựa chọn nhiều tài khoản khi rút hoặc gửi tiền, dịch vụ ủng hộ từ thiện, thanh toán điện tử.
3. Phát triển ứng dụng di động để người dùng có thể sử dụng hệ thống trên điện thoại di động.
4. Tối ưu hoá hiệu suất của hệ thống, bằng cách tăng cường khả năng xử lý dữ liệu và tối ưu hoá cấu trúc cơ sở dữ liệu.

5. Cải thiện giao diện người dùng và tăng tính thẩm mỹ của hệ thống để thu hút người dùng sử dụng.

Ngoài ra, sản phẩm còn có thể được cải tiến thêm các tính năng bổ sung khác để nâng cao trải nghiệm người dùng, ví dụ như cho phép người dùng chọn ngôn ngữ, hệ thống có thể được kết nối với các ngân hàng khác để cung cấp cho người dùng các dịch vụ chuyển khoản liên ngân hàng và các sản phẩm tài chính khác như thẻ tín dụng, khoản vay và tiết kiệm.

Tuy nhiên, với các tính năng hiện có, sản phẩm đã đáp ứng được các yêu cầu cơ bản và có thể được sử dụng để mô phỏng một hệ thống ATM thực tế. Sản phẩm có tiềm năng để phát triển và mở rộng, đáp ứng nhu cầu ngày càng đa dạng của người dùng về các dịch vụ tài chính.

TÀI LIỆU THAM KHẢO

- [1] B. Schneier, Applied Cryptography, Hoa Kỳ: John Wiley & Sons., 1994.
- [2] Ron Rivest, Adi Shamir và Leonard Adleman, RSA Encryption Algorithm, Communications of the ACM (Association for Computing Machinery) số 7/1978, 1977.
- [3] N. I. o. S. a. T. (NIST), Secure Hash Standard, the National Institute of Standards and Technology (NIST), 1993.
- [4] "Hướng dẫn học SQL cho người mới bắt đầu với MySQL," [Online]. Available: <https://openplanning.net/10237/huong-dan-hoc-sql-cho-nguoimoi-bat-dau-voi-mysql>.
- [5] "PreparedStatement trong java," [Online]. Available: <https://viettuts.vn/java-jdbc/preparedstatement-trong-java>.